

ATARI
ST
COMPUTER
Die Fachzeitschrift für den ATARI-ST Anwender.
September 90
DM 8,-
Ös. 64,-
Sfr. 8,-
9
Bildverarbeitung mit dem ATARI ST
• Retouche Professional
• TmS Cranach
Publishing Partner Master
DTP aus Übersee
Drucker der neuen Generation
ATonce
AT-Emulator
Hardwareprojekt
Zweite serielle Schnittstelle
GEM-Programmierung
Menüs in Fenst


MMAD

Konquistador ST

Zurück von der Sommerpause möchten wir uns in dieser Ausgabe unter anderem dem Thema "Elektronische Bildverarbeitung" widmen. Auf diesem Computer-Fachgebiet ist hauptsächlich der Apple Macintosh vertreten, der mit einer Vielzahl von leistungsfähigen Programmen eindeutig als Spitzenreiter gelten dürfte. Die Branche ist vor allen Dingen lukrativ, da viele Dinge, die bisher noch aufwendig von Lithografie-Anstalten erstellt werden mußten, nun recht billig selbst produziert werden können. Natürlich hat das gewisse Grenzen und verlangt auch einiges an Erfahrung.

Doch auch der ATARI ST ist derzeit in Begriff, einen Teil des Goldkuchens anzuknabbern. Mit Programmen wie z.B. Retouche Professional oder TmS Cranach läßt sich schon eine Menge in bezug auf Elektronische Bildverarbeitung anfangen. Derzeit muß Calamus noch als Schnittstelle zu professionellen Satzbelichtern herhalten, doch zumindest im Hause 3K wird demnächst eine eigene Ansteuerung für Retouche an einen Satzbelichter erhältlich sein. Mit Calamus SL kehrt auch in den DTP-Bereich die Farbe ein. Man sieht also: Auch mit dem ST darf man in Zukunft in dieser Mac-Domäne rechnen.

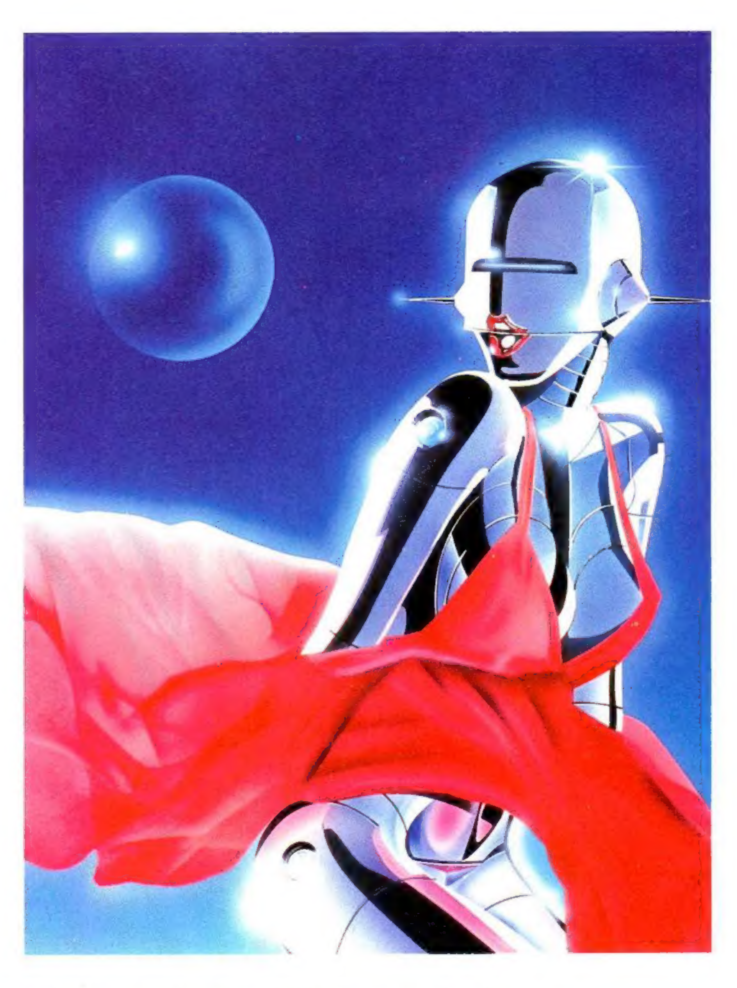
Problematisch bleibt lediglich der Rechner selbst. Doch auch hier kann man mit Arbeitsspeichererweiterungen, Grafikkarten, höheren Taktraten und anderen Prozessoren rechnen. Nicht zuletzt scheinen sich ja die Gerüchte über den neuen TT mit 32 MHz zu bestätigen, so daß sich auch dieses Problem lösen wird. Man darf also gespannt sein, was die Zukunft und vor allem die ATARI-Messe in Düsseldorf bringen wird.

Harald Egel

I N H A L T

SOFTWARE

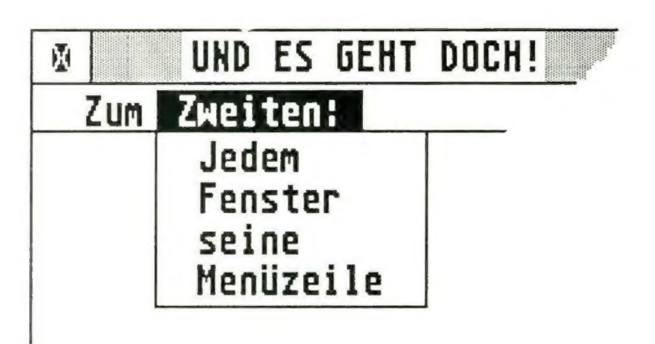
GRUNDLAGEN



Elektronische Bildverarbeitung mit dem ATARI ST

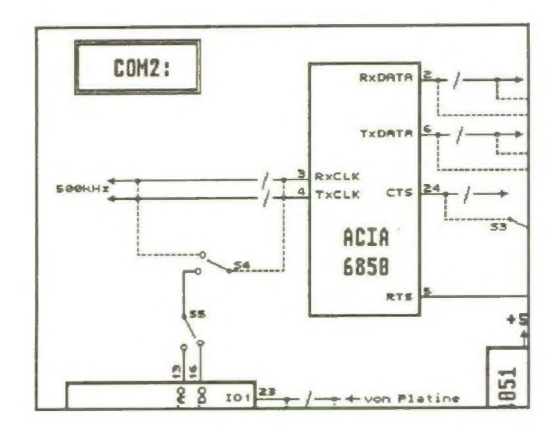
Immer mehr Atarianer entdecken die Welt der Elektronischen Bildverarbeitung (kurz EBV). Das nötige Werkzeug, um ordentliche Bildreproduktionen und -manipulationen vornehmen zu können, ist inzwischen auch vorhanden. Zwei Programme, TmS Cranach und Retouche Professional, möchten wir Ihnen hier vorstellen. Vielen Einsteigern fehlt allerdings auch das dringend notwendige Hintergrundwissen zur EBV. Wir wollen mit einem Grundlagenartikel versuchen, Ihnen den Einstieg zu erleichtern.

Seite 20, 26 und 31



Daß man mit "offiziellen" und "legalen" Mitteln unter GEM wesentlich mehr erreichen kann, als es sich die Entwickler von GEM vielleicht haben träumen lassen, ist inzwischen reichlich bekannt. Wie unser Titel schon verrät, spielen die Menüs bzw. die Menüzeile in diesem Artikel eine große Rolle, wenn es mal wieder gilt, die letzten Geheimnisse zu entschlüsseln.

Seite 100

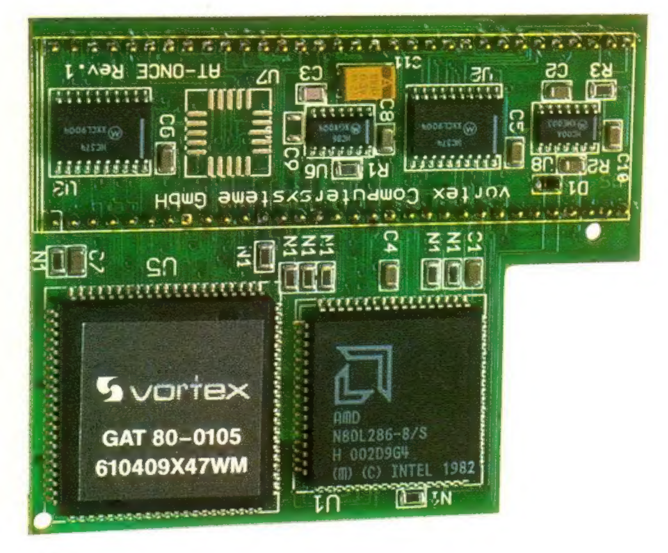


METAMORPHOSE

VOM MIDIPORT ZUR SERIELLEN SCHNITTSTELLE

Sicher hat sich schon mancher ST-Anwender geärgert, daß sein ST nur eine RS-232-Schnittstelle hat - der TT dagegen gleich vier davon. Doch deshalb brauchen Sie nicht gleich auf den TT umzusteigen. In Verbindung mit dem MIDI-ACIA läßt sich am ST relativ einfach eine zweite serielle Schnittstelle (COM2) realisieren. Zwar muß man dabei in Kauf nehmen, daß der MIDI-Port nicht angesprochen werden kann, solange man die zweite RS-232 benutzt, aber wer spielt schon auf seinem Keyboard und macht zur gleichen Zeit DFÜ?

Seite **129**



ATonce
- Der ST als AT

Mit Sicherheit darf man den ATARI ST mittlerweile zu einem der vielseitigsten Computer unserer Tage rechnen. Zu diesem Ruf haben die verschiedenen Emulatoren für den ST kräftig beigetragen. Macintosh, QL, PC-XT und jetzt auch PC-AT zeigen die hohe Flexibilität des ST. Den AT-Emulator ATonce von vortex möchten wir Ihnen in unserem Artikel vorstellen. Was er wirklich leistet, lesen Sie auf

Seite **191**

ST-REPORT

PROGRAMMIERPRAXIS

HARDWARE

AKTUELLES

RUBRIKEN



Portalog für den Portfolio



Mit dem Portalog bietet die Firma IBP ein sehr kompaktes, wirtschaftliches und netzunabhängiges Meßsystem an. Als Basis dient der Portfolio. 10 Meßkanäle stehen zur Verfügung, von denen einer auch Autoranging beherrscht. Spannungen bis 400 V AC/DC und Ströme bis 400 mA können direkt gemessen werden, Der direkte Anschluß beliebiger Sensoren ist ohne großen Aufwand möglich. Der Abstand für eine Aufzeichnung von Meßwerten ist von ms bis zu Tagen variierbar. Dabei lassen sich bis zu 250000 Meßwerte im System speichern. Verwendung findet das System überall dort, wo intelligentes oder mobiles Messen erforderlich ist. Der Preis für das Komplettsystem liegt bei DM 1688,-.

IBP Elektronik GmbH
Lilienthalstraße 13
3000 Hannover 1
Tel. (0511) 630963

Biodata-News

Die Installation der Netzwerk-Software wurde in der Version 3.0 durch Installations- und Konfigurationsprogramme wesentlich vereinfacht. Der Boot-Prozeß kann nun benutzer- und knotenabhängig gestaltet werden. Neben dem bekannten dedizierten MS-DOS-Server, der jetzt auch unter Windows 2.0 läuft, bietet Biodata nun eine günstige Alternative zum Einstieg in Ethernet an: den Atari ST als nicht dedizierten Server, an ihm kann also während des Netzwerkbetriebs gearbeitet werden. Bis zu 5 Arbeitsplätze lassen sich so mit 10 MBit/s vernetzen.

Neu unter BioNet 100 sind MS-DOS-Rechner und der Apple Macintosh als Clients, was den beliebigen Austausch von Daten zwischen diesen und dem Atari ST ermöglicht. Selbstverständlich können auch mit diesen Clients Mail und Grafiken verschickt bzw. empfangen werden. Ebenfalls neu im Netzwerk ist der SuperCharger von Beta Systems. Für den ST bietet Biodata ein X-Window-System an, das den ST als Display-Server nutzt. Es handelt sich um eine Implementation der originalen MIT-Sourcen Version 11.3, die sowohl als Programm als auch als Accessory installiert werden können. Interessant ist in diesem Zusammenhang, daß "X" unter BioNet 100 auch mit Großbildschirmen arbeitet.

Als Server kann inzwischen auch eine beliebige UNIX-Maschine dienen (nicht nur 80386-Rechner, sondern frei wählbar, z.B. eine SUN), auf der dann die Serversoftware wie eine normale Task im Hintergrund abgearbeitet wird. Interessant für Netzwerke, die auf höchste Sicherheit Wert legen: der Parallel-Server. Neben dem normalen MS-DOS-Server wird ein zweiter Server installiert, der exakt die gleichen Arbeitsschritte vornimmt. Sollte durch unglückliche Zufälle einer der beiden Server ausfallen, erhält der Superuser eine Meldung, während der Netzwerkbetrieb auf dem anderen Server ohne Unterbrechung weitergeht. Alle Neuerungen werden auf der Atari-Messe in Halle 12 zu sehen sein.

Biodata GmbH
Flughafen Siegerland
5909 Burbach
Tel. (02736) 5152

Querdruck2 nun auch für den Laser

Beliebige ASCII-Texte lassen sich mit Querdruck2 ab jetzt auch auf dem Atari-Laser um 90° gedreht ausdrucken. Dabei erhält der Benutzer eine übersichtliche Oberfläche, komfortable Seitengestaltung, mehr Schriftgrößen und -arten, einen eigenen Zeichensatzeditor, grafische Zeichen aus dem IBM-Zeichensatz und vieles mehr. Das Hilfsprogramm für Tabellenkalkulationen, Datenbanken und Textverarbeitungen wird inklusive ausführlicher Anleitung und Tutorial ausgeliefert und kostet DM 78,-.

Entwicklungsbüro Dr. Ackermann
Kanalweg 1a
8048 Haimhausen
Tel. (08133) 1053

```
ACC   ansehen drucken  Font   Rand  W-Texte Sonder  laden  sichern  Ende

Drucker-Typ      Blattbreite       Zeilenabstand    Text-Datei : JOURNAL.TXT
[-] 24-N [+]     [-] A4 [+]Zoll    [-]  5 [+]       Zeichensatz: 024FETT3.FNT
                     203 mm        ca 1.0zeilig

Schriftgröße     Blattlänge        Sperrung         Modus
[-]15 CPI[+]     [-] 2135[+]Zoll   [-]  0 [+]       [-]WP-Model[+]
                     54221 mm           0 %

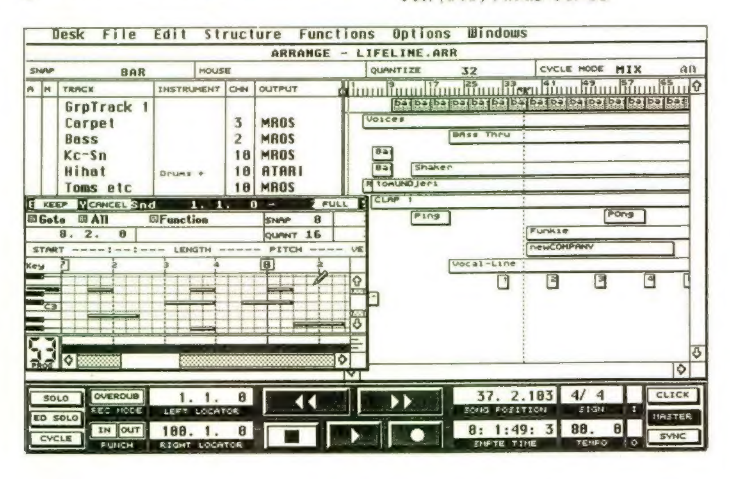
     [-] 10[+]     Maximale Zeilenlänge pro Blatt: 32000     [-] 10[+]

[-] 2[+]    Rand oben
[-] 0[+]    Kopfzeilen
[-] 0[+]    Überschrift
         Rand links        QUERDRUCK2          Rand rechts   Nächstes Blatt
  71        Textbereich
[-] 0[+]    Fußzeilen
[-] 2[+]    Rand unten
```

Neu von Steinberg: CUBEAT

Mit CUBEAT bietet Steinberg ein umfangreiches MIDI-Recording-Programm für den ST an. CUBEAT verfügt über 64 Aufnahmespuren. Es können insgesamt bis 16 Songs im Computer bearbeitet werden. Wie alle neueren Steinberg-Produkte arbeitet auch CUBEAT auf der Grundlage des neuen Multitasking-Betriebssystems M.ROS. CUBEAT bietet dem Anwender die gleiche leicht erlernbare grafische Benutzeroberfläche, die auch schon CUBASE bei Profis und Einsteigern populär gemacht hat. CUBEAT gibt eindrucksvoll die Struktur der Komposition wieder. Schnell und direkt läßt sich ein Arrangement mit grafischen Werkzeugen auf dem Bildschirm edieren. Einzelne MIDI-Events lassen sich im Key- und Grid-Editor bearbeiten. CUBEAT ist kompatibel zum Steinberg SMP 24, TIMELOCK, MIDEX und MIDEX plus und läuft auf allen STs mit mindestens 1 MB RAM, s/w- oder Großbildschirm. Der Preis beträgt DM 490,-.

Steinberg GmbH
Billwerder Neuer Deich 228
2000 Hamburg 26
Tel. (040) 78985-16/-66



MBF

Das Fakturierungs- und Warenwirtschaftssystem MBF ist eine Kunden, eine Artikel-, Lagerverwaltung und vieles mehr. Lieferanten, Vertreter und Werkstatt können ebenso verwaltet werden wie die Historie für Kunden, Artikel und Werkstatt. Eine Textverarbeitung ist ebenso in das Programm integriert wie eine Provisionsabrechnung, Suchfunktionen, bis zu zwei Millionen Datensätzen pro Datei, 2211 Seiten im Programm integrierter Hilfstext uvm. Die Vollversion kostet DM 2223,-, eine Demo-Version gibt's kostenlos.

Thomas Hery
Thebäerstraße 8
5500 Trier
Tel. (0651) 28230

Gebt Viren keine Chance!

Mit diesem Slogan stellt Afusoft den AVSS (Afusoft Viren-Selbstschutz) vor. Mit ACSS sind ab September '90 alle Software-Produkte von Afusoft ausgestattet. Dieser bei professioneller Software bisher einmalige Virenwächter überprüft ständig sein Programm und schlägt Alarm, wenn es angegriffen wird. Der Benutzer ist gewarnt, kann reagieren und somit evtl. größere Schäden vermeiden. Allerdings übernimmt der Schutz nicht die Identifizierung und Lokalisierung von Viren, hier ist weiterhin der Anwender gefordert.

Afusoft
Postfach 2113
7535 Königsbach-Stein 2
Tel. (07232) 1862

CIS-Lohn&Gehalt gesamtdeutsch

Entsprechend der aktuellen politischen Großwetterlage hat sich die Firma Ciechowski in den letzten Wochen und Monaten darum bemüht, ihr Progamm CIS-Lohn&Gehalt an die Erfordernisse der DDR anzupassen. Nicht zuletzt durch die Währungs- und die zum 1.1.91 in Kraft tretende Steuerunion ist es unbedingt erforderlich, daß die - jetzt auch neu gegründeten - Betriebe der DDR auf einfache Art in die Lage versetzt werden, die über Nacht kompliziert gewordene Lohnabrechnung zu bewältigen. Besteller aus der DDR erhalten daher - zunächst befristet bis zum 31.12.90 - einen Preisnachlaß von 10%.

Ciechowski Computer Innovations
Ober-Saulheimer Straße 18
6501 Wörrstadt
Tel. (06732) 7354 o. 5018

Platon 1.4

Die neue Version von Platon, die auf der Atari-Messe zur Vorstellung kommt, wird vollständig objektorientiert arbeiten. Ein 14poliges IC wird dann wirklich als ganzes IC und nicht wie 14 einzelne Lötpunkte behandelt. Durch die Gruppierung von ganzen Plattenbereichen läßt sich die Arbeit schneller und komfortabler gestalten. Des weiteren enthält das Programm einen GDOS-Treiber, mit dem z.B. Metafiles erzeugt werden können. So kann man z.B. Platinen in Calamus einlesen. Auch die Bedieneroberfläche wurde verbessert (Pop-Up-Menüs etc.). Außerdem können jetzt Leiterplatten zur schnellen Prototypenfertigung auf einer XYZ-Anlage umfräst werden. Kreis- und Flächenfüllfunktionen (z.B. für Masseflächen) sind ebenfalls hinzugekommen. Die Auflösung, in der die Platinen edierbar sind, wurde auf 1/2000 Zoll gesteigert. Das Programm ist erhältlich bei

VHF-Computer
Maurener Weg 115a
7030 Böblingen
Tel. (07031) 289211

Vortex - Es geht nahtlos weiter

In der letzten Zeit hörte man einige Gerüchte über den Konkurs der vortex Computersysteme GmbH. Am 1.6.1990 wurde die neue vortex Computersysteme GmbH gegründet. Das Stammkapital beträgt derzeit 0,6 Mio. DM. Bis Ende Juli haben weitere private Investoren zusätzliche Finanzmittel in Höhe von DM 200000,- bereitgestellt. Die Geschäftsführung der neuen vortex Computersysteme GmbH liegt für eine Übergangszeit ausschließlich in den Händen von Monika Armbruster. Der bisherige technische Geschäftsführer Reinhardt Michel wird diese Funktion im Zuge einer gesellschaftsrechtlichen Neuordnung nach Abschluß der Verhandlungen wieder einnehmen. Innerhalb der Produktpalette und der sonstigen Leistungen gibt es ebenfalls keinerlei Änderung. Der Verkauf der Festplatten-Subsysteme System 2000 sowie Data-Jet geht weiter. Der Konkurs konnte aufgrund von Unstimmigkeiten unter den Gesellschaftern über die zukünftige vortex-Geschäftspolitik und daraus resultierender akuter Finanzprobleme nicht abgewendet werden. Inzwischen ist aber im Hause vortex der Alltag wieder eingekehrt.

vortex Computersysteme GmbH
Falterstraße 51-53
7101 Flein bei Heilbronn
Tel. (07131) 5088-0



CW-Chart 8.0

Ab sofort liegt die neue Version 8.0 des Programms CW-Chart vor. Einige Neuerungen sind z.B. Signaltabellen (Point&Figure, Stochastik, Trendbestätigung, Phasenanalyse, Parabolic), kombiniert mit unterstützenden Indikatoren (Momentum, TBI, Overbought-Index, RSI...); Stochastik-Indikator und -Tabelle; Phasenanalyse und -tabelle; Optionstabelle mit Black&Scholes-Formel; Großbildschirmeignung; DTB-Optionskurse per BTX; zirka 180000 Kurse kostenlos. Der Upgrade-Preis beträgt DM 99,-, eine Demo-Disk ist bereits für DM 20,- erhältlich bei

Foxware
Buchsteinweg 1
8172 Lenggries
Tel. (08042) 2175

Neuer Preis - Okilaser 400

Ab dem 1. September 1990 ist der neu empfohlene Verkaufspreis von DM 2998,- (vorher: DM 3798,-) für den Okilaser 400 gültig. OKI führt diese Preissenkungsmaßnahme aus, um den stetig steigenden Marktanforderungen und Anwenderbedürfnissen verstärkt gerecht zu werden und leistet einen großen Beitrag in die für kostenbewußte Anwender richtige Richtung. Es erfolgt jedoch am Okilaser 400 keine Einbuße an Ausstattungsmerkmalen. Der Drucker ist voll kompatibel zur HP LJ-Serie II, IBM Proprinter XL und Diablo 630.

Okidata GmbH
Hansaallee 187
4000 Düsseldorf 11
Tel. (0211) 59794-0

hyperCACHE-030

Der seit Anfang des Jahres angekündigte Beschleuniger für Mega STs mit einem 68030-Prozessor "hyperCACHE-030" wird ab Anfang September erhältlich sein. Die von der pro_VME in Neckargemünd entwickelte 6-Lage-Platine enthält einen MC68030-Prozessor (25 MHz), 16 kB externen Cache-Speicher, einen Sockel für die FPU und das an den MC68030 angepaßte GEM/TOS, untergebracht in 4 EPROMs. Der interne Bus ist 32 Bit breit. Aus Kompatibilitätsgründen besteht die Möglichkeit, auf den 68000-Prozessor zurückzuschalten. Der Mega-Bus bleibt weiterhin frei für andere Erweiterungen. hyperCACHE-030 wird auf der Atari-Messe zu sehen sein und kostet in der Grundversion DM 2498,-.

pro_VME
Bahnhofstraße 44
6903 Neckargemünd 1
Tel. (06223) 72029

ReProK: Schnittstelle fibuMAN, Update

Auf Wunsch vieler Interessenten wird Stage Microsystems zur Atari-Messe '90 für das Bürosystem ReProK eine weitere Fibu-Schnittstelle anbieten. ReProK kann somit an die Finanzbuchhaltungen TiM und fibuMAN angeschlossen werden. Nach Festlegung des Kontenrahmens werden die Buchungsdaten nach dem Verlassen von ReProK an die Finanzbuchhaltung übergeben und automatisch aufbereitet. Die neue Version (Rel. 079001) kann wieder mit einer Vielzahl funktionaler Verbesserungen aufwarten, die auf der Atari-Messe zur Präsentation kommen sollen.

Stage Microsystems
Lohmühler Berg 30
5620 Velbert 15
Tel. (02053) 3179

Komfortable Textkorrektur

Mit Lektorat bietet WAVE Computersysteme in Gießen eine leistungsstarke Textkorrektur an, die über ASCII- und Wordplus-Textformate hinausgehend auch Signum!2-Texte direkt einlesen kann. Ein vom Anwender erweiterbares Standardlexikon mit 110000 Wörtern sowie ein Konfigurationsprogramm, mit dem alle Menüeinträge und Dialogboxen auch über die Tastatur gesteuert werden können, sind im Lieferumfang bereits enthalten. Lektorat kann mit maximal 14 weiteren vom Anwender selbst erstellten Lexika bei jedem Korrekturlauf arbeiten. Lektorat läuft auf allen STs und unter allen TOS-Versionen ebenso wie auf allen Großbildschirmen. Das Programm kostet DM 149,- und ist zu beziehen über

WAVE Computersysteme GmbH
Südanlage 20
6300 Gießen
Tel. (0641) 72357

Textstatistik

Textname: PROBE1.DOC

Seitenanzahl: 5
Zeilenanzahl: 154
Zahl der Absätze: 25
Anzahl der Sätze: 58
Anzahl der Worte: 753
mittl. Wortlänge: 6.23
mittl. Satzlänge: 12.98
Silbentrennungen: 2

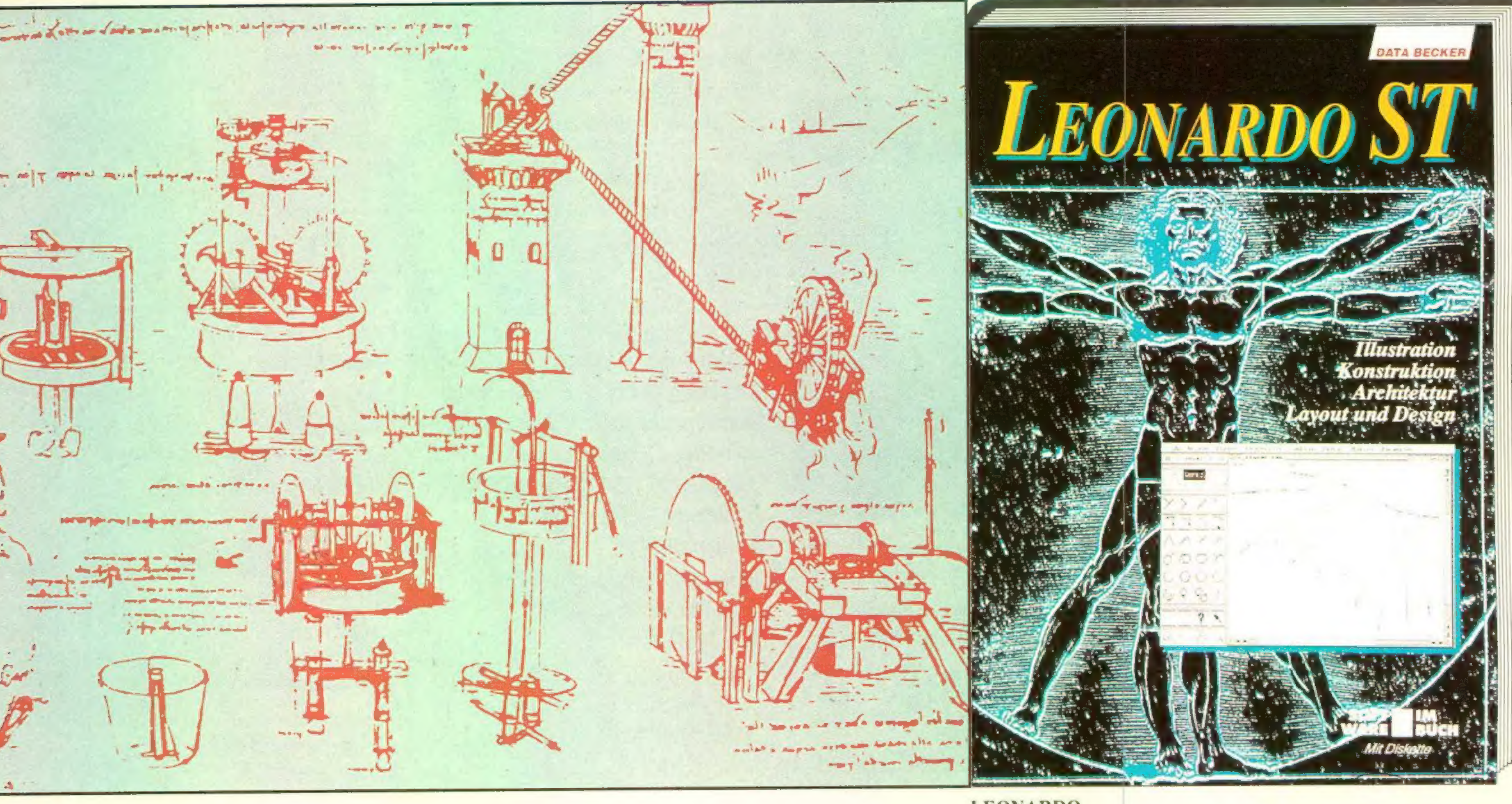
DIAGRAMM
Wortlängen
Satzlängen
Ok

AutoSwitch-OverScan

In der letzten Ausgabe ist leider der Preis für die Bildschirm-Auflösungserweiterung OverScan falsch abgedruckt worden. Der korrekte Preis ist DM 120,-. Hinzu kommen DM 7,- für Porto und Versand bei Nachnahme bzw. DM 5,- für Porto und Versand bei Vorkasse oder Verrechnungsscheck.

OverScan GbR
Säntisstraße 166
1000 Berlin 48
Tel. (030) 8115882

Die Sensation der ATARI-Messe!



LEONARDO
DM 99,-
ISBN 3-89011-818-6

Geniale Software zu einem großartigen Preis – das hat auf der ATARI-Messe Furore gemacht. Für alle, die nicht dabei sein konnten, hier die Kurzvorstellung: LEONARDO ist die Software, mit der Sie Ihre Ideen umsetzen können!

Durch LEONARDO verbinden Sie mathematische Strenge mit künstlerischer Freiheit – inklusive der Freiheit, sich das Programm auf eigene Bedürfnisse zuzuschneiden. Lassen Sie sich beflügeln durch Funktionen in Hülle und Fülle: LEONARDO unterstützt Sie bei mathematischen, architektonischen und grafischen Aufgaben unter anderem durch objektorientiertes Zeichnen in allen Variationen mit hoher Präzision; verschiedene Kurvenelemente für den Freihand-Entwurf; 250 Zeichenebenen; eine neuartige, sich dem Anwender anpassende Benutzeroberfläche; eine Undo-Funktion, mit der Sie alle (!) Zeichenvorgänge Schritt für Schritt widerrufen können; umfangreiche Bemaßungsfunktionen und Werkzeuge zum Konstruieren. Natürlich können Sie auch Flächen mit Schraffuren oder Mustern füllen, Symbole erstellen und einfügen sowie die Vektor-Schriften für Illustrationen nutzen.

LEONARDO erhalten Sie als Software im Buch. Das Programm unterstützt gängige 9- und 24-Nadeldrucker, PostScript-Drucker, HP-kompatible Plotter und Großbildschirme. Mit LEONARDO erstellte Grafiken lassen sich in alle Programme übertragen, die GEM-Metafiles einbinden, wie z. B. Calamus, Timeworks, GEM-Draw u. a. Mindestkonfiguration: ATARI ST ab 1 MByte.

LEONARDO ST – damit Sie Ihre Ideen umsetzen können!

DATA BECKER

Bitte einsenden an: DATA BECKER, Merowingerstraße 30, 4000 Düsseldorf 1

Wer zuerst kommt, malt zuerst!

Schicken Sie mir sofort das innovative Zeichenprogramm ☐ LEONARDO

Ich bezahle ☐ per Nachnahme ☐ mit beiliegendem Verrechnungsscheck

Name

Straße

PLZ/Ort

CSS-Gigafile 650

Die CSS-Gigafile 650 hat ab dem 1.8.1990 einige zusätzliche Leistungsmerkmale aufzuweisen: diverse technische Aufwertungen, 1 Jahr Garantie, TÜV/GS2-Prüfsiegel, Papst-Lüfter, SCSI 25- oder 50polige, anschlußfertige Auslieferung für DM 9980,-, Anschluß-Kit für Atari ST-ICD-Controller mit Software und CSS-Software für DM 398,-. Außerdem existiert ab sofort ein technischer Hotline-Service für eingetragene Kunden, das Gerät kann gegen Aufpreis auch vor Ort installiert werden.

Computer-Systeme Suplie
Landwehr 53
4670 Lünen
Tel. (02306) 52489



AT-Speed - Neues Update V 2.1

Unter AT-Speed - dem MS-DOS Hardware-Emulator der Sack Electronic GmbH - kann ab sofort der ATARI Laserdrucker genutzt werden. Auch WINDOWS 3.0 im Protected Mode - dem neuen Betriebssystem-Standard - läuft mit AT-Speed.

Die bei AT-Speed bestehenden Möglichkeiten des schnellen Wechsels zwischen TOS und DOS mittels ACCESSORY bleiben dabei erhalten.
Die Update-Version 2.1 mit allen neuen Erweiterungen ist beim ATARI Händler erhältlich.

Computerclubs

Ab sofort sammelt die Redaktion wieder Adressen von Computerclubs, die regelmäßig in einer eigenen Rubrik veröffentlicht werden sollen. Daher die Bitte an alle Clubs, uns zahlreich mit den folgenden Angaben zu schreiben: Computertypen, Beiträge, Leistungen, Schwerpunkte, Anschrift. Den Anfang macht diesmal der neugegründete B.I.T.S.-Computer-Club Berlin. Alle Besitzer von MS-DOS-, Atari ST-, Amiga- oder C64-Geräten können mitmachen. Die Leistungen des Clubs: eine monatliche Zeitschrift, Clubraum, PD-Service, Spielturniere, Demo-Service, Wettbewerbe, Clubrabatte bei Software und Zubehör. Die Schwerpunkte: Einsteigerhilfen, Tips&Tricks, Lösungen, Erfahrungsaustausch. Ein Beitrag ist nicht zu entrichten.

B.C.C. Berlin
Jagowstraße 17
1000 Berlin 21
Tel. (030) 3938203

MegaPaint II mit FAX-Ansteuerung

Mit den beiden neuen Produkten MegaPaint II Professional und MegaPaint II PC von Tommy-Software aus Berlin ist man in der Lage, Fernkopierer via FAX-Box anzusteuern. Voraussetzung dafür ist der Erwerb einer postzugelassenen FAX-Box. Diese Geräte werden z.B. von den Firmen "Music Mail Service" und "Touchbase Systems" angeboten.

TommySoftware
Selchower Straße 32
1000 Berlin 44
Tel. (030) 621406-3

OCR für unter DM 200,-

Die Marvin AG wird auf der Atari-Messe ihr neues OCR-Produkt vorstellen. SYNTEX läßt sich auf jede beliebige gedruckte, nicht gebundene Schrift trainieren. Trotz einer außergewöhnlichen Erkennungsrate von 99,9% bis 100% ist SYNTEX extrem schnell. Bis zu 120 Zeichen pro Sekunde oder 7200 Zeichen pro Minute werden erkannt. Die von OCR-Software so häufig gemachten Verwechslungen zwischen "1" und "I", "O" und "0" sowie "1" und "l" werden automatisch erkannt und korrigiert. SYNTEX paßt seine Parameter automatisch an die Qualität der Vorlage an. Der Preis (inklusive umfangreichem deutschem Handbuch) beträgt DM 198,-.

H. Richter
Hagener Straße 65
5820 Gevelsberg

ATonce: Stecken statt Löten

Beim Einbau des AT-Emulators ATonce kann das für manche Anwender lästige Löten entfallen - sofern ATonce in einen Mega ST oder 1040 STE eingebaut wird. Die Steckadapter für den Mega ST und 1040 STE sind ab sofort bei den vortex-Fachhändlern erhältlich. Der Steckadapter für den Mega ST kostet DM 98,-, für den STE müssen DM 128,- bezahlt werden. So dauert der Einbau nur noch fünf Minuten. ATonce ist auch auf allen Speedbridges (Steckadapter der Firma digital image), die nach dem 1.6.1990 ausgeliefert wurden, ohne Einschränkung lauffähig. Hier ist beim Einbau auf jeden Fall darauf zu achten, daß die Meßklemme auch an Pin 23 der CPU angeklemmt wird.

digital image
Postfach 1206
6096 Raunheim

vortex Computersysteme GmbH
Falterstraße 51-53
7101 Flein

Datenschutz für Platten und Disketten

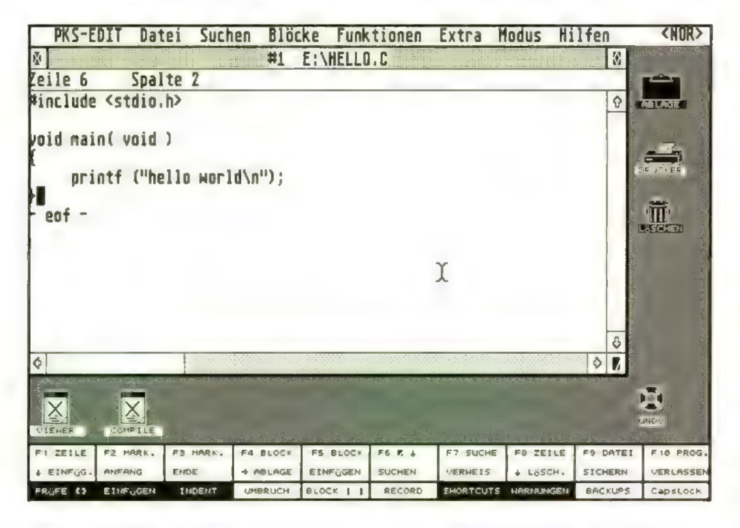
Datenschutz auf dem ST für Festplatten, Disketten und sonstige Massenspeichermedien garantiert 1st_Lock, ein neues Programm von LogiLex. Die erste Stufe des Schutzes besteht in dem Zwang, nach dem Einschalten des Rechners bzw. nach Arbeitsunterbrechung ein Paßwort einzugeben, um auf das Medium zugreifen zu können. Die zweite Stufe ist während der Arbeit mit anderen Programmen im Hintergrund aktiv: Die zu speichernden Daten werden während des Speichervorgangs ver- und während des Ladevorgangs entschlüsselt. Gleichzeitig werden Directories und FAT codiert. Solange 1st_Lock aktiv ist, können sie jedoch normal gelesen werden. Diese Verschlüsselung ist für jede(s) Laufwerk/Partition einzeln zu- bzw. abschaltbar. In einer dritten Stufe wird der Schutz durch individuelle Verschlüsselung erreicht: Jedes 1st_Lock hat einen unterschiedlichen Code, der einen Zugriff auf die Daten durch Dritte, die ebenfalls 1st_Lock verwenden, verhindert. Dadurch können selbst bei Diebstahl des Datenträgers weder Daten und Programme, noch Programmnamen, Dateilängen usw. ausspioniert werden. 1st_Lock gibt es in einer Single- (DM 189,-) und in einer Multi-User-Version, die bis zu 32 Benutzern durch individuell änderbare Paßwörter gezielt Lese- bzw. Lese-/Schreib-Zugriffsrechte für die verschiedenen Laufwerke einzeln zuweisen kann (DM 498,-).

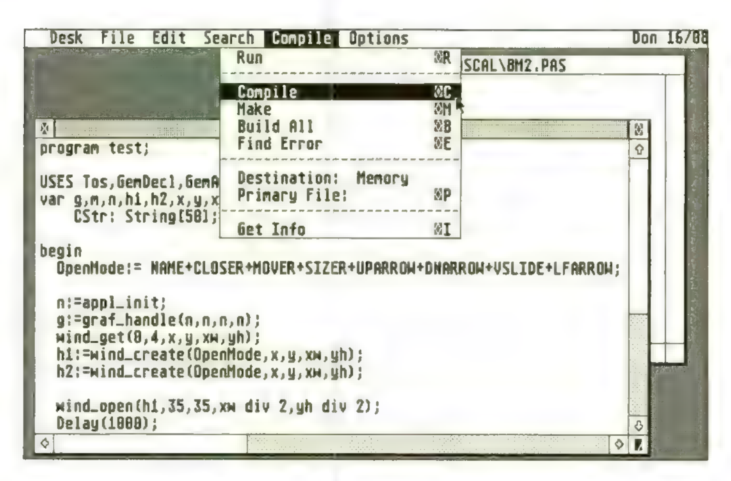
LogiLex - G. Oppenhorst
Eifelstraße 32
5300 Bonn 1
Tel. (0228) 658346

Neues von PKS

Von PKS, dem Entwickler von PKS-Write, gibt es einen neuen Texteditor für gehobene Ansprüche. In diesem schnellen GEM-Editor stehen aus der UNIX-Welt bekannte Funktionen wie reguläre Ausdrücke für die Definition beliebiger Suchmuster und Verweise zur Verfügung. Komplette Disketten oder Pfade können durchsucht werden. Textblockmanipulationen sind auch in Spaltenform möglich. Undo arbeitet für alle Funktionen. Zur Bearbeitung von Dateien mit beliebigem Format (sogar Binärdateien) können Dokumenttypen definiert und mit einem Satz von Bearbeitungseinstellungen verknüpft werden. Selbstverständlich enthalten sind freie Tastenbelegungen, Makros, Abkürzungen und Ereignisrekorder! PKS-EDIT arbeitet mit jedem Compiler zusammen und springt bei fehlerhaften Programmtexten automatisch in die entsprechenden Zeilen. Ideal ist der Editor auch für TeX-Benutzer. Eingebaut ist eine Shell mit vielen Funktionen, für gehobene Ansprüche ist eine erweiterte Bourne-Shell mit MAKE und vielen Dienstprogrammen verfügbar. Ausdruck von Dateien auch im Wordplus-Format (Druckertreiber voll kompatibel!). Das Programm wird zum Preis von DM 148,- angeboten. Eine Demodiskette und Informationen sind erhältlich bei

Pahlen & Krauß Software
Dieffenbachstraße 32
1000 Berlin 61
Tel. (030) 7865945





Neues PASCAL-Entwicklungssystem von MAXON

Brandneu zur ATARI-Messe präsentiert sich MAXON-PASCAL, ein leistungsfähiges, vollintegriertes Programmentwicklungspaket für ATARI ST/TT. Es beinhaltet GEM-Editor, Compiler und Linker und ist sowohl in den Anwenderfunktionen als auch im Programm-Code voll kompatibel zu Turbo Pascal 5.0 auf dem PC. Der kompakte, schnelle Single-Pass-Compiler belegt wenig Speicherplatz, wodurch die Bearbeitung größerer Programme erleichtert wird. Compilierungsgeschwindigkeit: mehr als 20000 Zeilen/min. auf einem normalen ST. Es besteht die Möglichkeit, wahlweise auf Disk oder im Speicher zu arbeiten. Compiler und Linker werden vom Multi-Window-Editor aufgerufen und benötigen daher keine Ladezeit. Durch das von Turbo Pascal 5.0 bekannte UNIT-Prinzip wird ein einfacher, modularer Aufbau von Programmen möglich. Als Standard-Units werden mitgeliefert: GEM/VDI, TOS, BIOS, ST-Pascal, UTIL und PRINTER. Oft benötigte Units können zu Libraries zusammengefaßt werden. Eine komplexe und funktionsabhängige Hilfsfunktion unterstützt den Anwender in Fragen auf die PASCAL-Syntax, den Editor und die generelle Bedienung. Der unverbindliche Verkaufspreis beträgt DM 259,-.

Ebenfalls neu ist ein schneller PROLOG-Interpreter mit eingebautem Editor und kompletter Edinburg-Syntax. Er verfügt über eine Geschwindigkeit von ca. 2500 Lips, ca. 120 eingebaute Primitive und 300 in PROLOG geschriebene Prädikate der Entwicklungsumgebung. AES/VDI-, GEMDOS-, BIOS- und XBIOS-Aufrufe sowie Peek und Poke im geschützten Bereich sind möglich.

MAXON Computer
Schwalbacher Str. 52
6236 Eschborn
Tel. (06196) 481811

80C196 für den ST

Ab sofort gibt es ein vollkommen GEM-gesteuertes Entwicklungs-Kit für Intels 80C196-Microcontroller. Geliefert werden eine Entwicklungskarte im Euro-Format, Software (Assembler, Linker), 130seitiges Handbuch und diverse Hilfsprogramme zum Preis von DM 498,-.

Siegried Cordes
Hochlandweg 3
8153 Neukirchen
Tel. (08020) 1490

FSE-Festplatten nun mit SCSI-Schnittstelle

Ab sofort werden alle FSE-AHS-Festplatten serienmäßig mit einer SCSI-Schnittstelle angeboten. Der Bus ist an der Rückseite mit einem 50poligen Centronics-Stecker herausgeführt. Somit können zusätzliche SCSI-Erweiterungen wie z.B. Streamer, Wechselplatte etc. ohne Öffnung des Gehäuses an die Festplatte angeschlossen werden.

Frank Strauß Elektronik
Schmiedstraße 11
6750 Kaiserslautern
Tel. (0631) 67096-98

Shift-Neuheiten

Shift wird dieses Jahr gleich zweimal auf der Atari-Messe vertreten sein. Zum einen auf dem Sonderstand DTP, der dieses Jahr einen besonderen Messe-Schwerpunkt bildet. Vorgestellt wird ConVektor, ein Vektorisierungsmodul für das Grafikprogramm Arabesque, mit dem es möglich ist, Rastergrafiken automatisch in Vektorgrafiken umzuwandeln. Von Arabesque selbst wird die Version 1.20 zu sehen sein, die mit Bezierpolygonen arbeitet und damit noch besser für den professionellen Einsatz geeignet ist. Zum anderen zeigt Shift als absolute Neuvorstellung die komfortable und schnelle Textverarbeitung Cy-Press, die im Oktober auf den Markt kommen soll. CyPress arbeitet mit Proportional- und Drucker-Fonts, beliebig großen Grafiken und stellt ein Korrektursystem von Langenscheidt zur Verfügung.

Shift
Unterer Lautrupweg 8
2390 Flensburg
Tel. (0461) 22828

600 dpi Laserkit

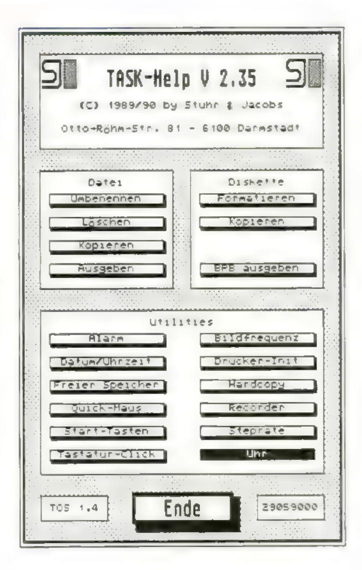
Der Entwicklungsabteilung der Marvin AG aus Zürich ist ein kleines Meisterstück gelungen: Mittels einer außergewöhnlichen Aufrüstung vermag ein herkömmlicher Atari-Laserdrucker statt 300x300 dpi satte 600x300 dpi zu drucken. Und dies nicht zu den üblichen MS-DOS-Umrüstpreisen ab DM 3000,-, welche das gleiche Verfahren anwenden, sondern für DM 320,-. Die Qualitätsverbesserung ist sogar mit bloßem Auge verblüffend gut sichtbar. Kurven und schräge Linien werden präziser gezeichnet. Selbstverständlich ist die Auflösung umschaltbar, damit keinerlei Probleme entstehen, falls das eine oder andere Programm noch nicht angepaßt sein sollte. Der Lieferumfang umfaßt eine ausführliche Einbauanleitung, eine Treiber- und Diagnosediskette sowie das dpi Controlling Board.

Marvin AG
Friesstraße 23
CH-8050 Zürich
Tel. (00411) 3022179

Task-Help

Bei Task-Help handelt es sich um ein "Multi-Accessory". Mit ihm können Dateien umbenannt, gelöscht, kopiert oder ausgegeben. Disketten kopiert und formatiert werden, vier Alarmzeiten sind vorhanden, Datum und Uhrzeit lassen sich einstellen, der freie Speicherplatz wird angezeigt, die Step-Rate kann verändert werden und viele Optionen mehr. Task-Help kostet DM 89,- und ist erhältlich bei

Stuhr & Jacobs
Otto-Röhm-Straße 81
6100 Darmstadt
Tel. (06151) 82885



eLAN für ATARI TT

Das Netzwerk-System eLAN der Firma GTI aus Berlin ist ab sofort auch für die neuen ATARI TTs verfügbar. Das dazugehörige Netzwerkbetriebssystem hört auf den Namen eLAN-TOS030. Als Hardware-Anschluß steht sofort das bewährte DMA-Subsystem oder in Zukunft eine VMEbus-Einbaukarte zur Verfügung. Im Netzwerk arbeitet ein TT dank 68030-Prozessor und Cache wesentlich schneller als ein Mega ST. Damit dürfte die Maschine besonders für Netzwerk-Anwender interessant sein, die von mehreren Arbeitsplätzen auf zentrale Datenbestände zugreifen wollen.

GTI GmbH
Unter den Eichen 108a
1000 Berlin 45
Tel. (030) 8315021

DRUCKERmens und FINAmens zur Atari-MESSEmens

DRUCKERmens ist eine Fakturierung für kleine bis mittlere Druck- und Satzbetriebe sowie Grafiker und Layouter, die komplette Druckaufträge abwickeln. Kunden-, Artikel- und Lieferantenstammdaten sind ebenso integriert wie Rechnungs- und Mahnwesen. Das Programm wird demnächst auch in einer ADItalk-Version geliefert. FINAmens ist ein Programm für Immobilienhändler, die bequem und komfortabel ihre Kunden und Immobilien verwalten wollen. Komplette Exposés sind ebenso möglich wie Mailing-Aktionen und Verwaltung der laufenden Verträge. Ein Modul zur Darlehensabrechnung ist in Vorbereitung. DRUCKERmens kostet DM 448,-, FINAmens DM 698,-.

comtex computersysteme
Gitteweg 3
7801 Bollschweil
Tel. (07633) 50784

Font-Serie "Leonardo Fontware GbR"

Aus dem Hause Leonardo Fontware kommen neuen Vektorgrafik-Fonts für Calamus. Diese Zeichensätze sind natürlich belichtungsfähig und in der Regel auch mit den für Calamus erhältlichen Zusatzprogrammen einsetzbar. Auf Wunsch ist jeder Font auch in condensed und kursiv erhältlich. Die Seriennummer des Programms wird nicht benötigt, da die Schriften durch eigene Seriennummern geschützt werden, d.h. jede Schrift ist auf jeder Version lauffähig. Die Fonts kosten DM 59,-. Bei Abnahme von 3 Fonts und mehr gibt es entsprechende Rabatte. Weitere Informationen gibt's bei

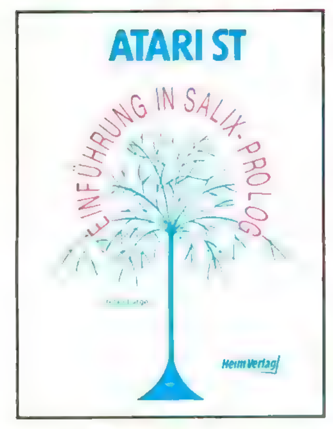
CARDPLAY
AltBerlin bold
AltBerlin reg
COMIC STRIP
Elan light
KINSLEY
Metro light
Metro bold
Pisa rounded
Malaga
Gate reg
Melody
PAINTCUT

Leonardo Fontware GbR
Hauptstraße 67
2905 Edewecht
Tel. (04405) 6809

Nachlese zum Technobox Drafter

In der Ferien-Vorfreude unserer Sommerausgabe und der damit verbundenen Hektik unseres Tests des TechnoBox Drafters sind uns leider Fehler unterlaufen. Bei diesem Zeichenprogramm ist gegenüber dem Technobox Cad/1 keine Funktion weggefallen. Es wurden lediglich einige Kopierbefehle zusammengefaßt. Ferner ist die Schraffur in allen Strich- und Linienarten möglich. Und die Art der Koordinatenanzeige läßt sich auch während der Konstruktion eines Elementes ändern. Der Hersteller kündigt für die ATARI-Messe die Version 2.0 des Drafters an, die über eine ganze Reihe von Funktionserweiterungen verfügen wird und in der die meisten Kritikpunkte unseres Tests behoben sein werden. Wir werden in unserem Messebericht aus Düsseldorf darüber berichten.

Buch-Neuheiten



Einführung in Salix Prolog

Hardcover
Best.-Nr. B-448
ISBN 3-923250-88-6
DM 49,- (unverbl. empf. Verkaufspreis)

Ein Fachbuch für ST-Besitzer, die sich für Künstliche Intelligenz interessieren. Mit einem leicht verständlichen Einstieg in deren wichtigste Programmiersprache: Prolog. Ziel des Buches ist es, vor den gängigen Fallstricken zu bewahren, die vor allem Umsteigern von anderen Sprachen drohen und fundierte Grundkenntnisse der Prologprogrammierung zu vermitteln.
Teil 1 erläutert, wie das Programm auf den verschiedenen Rechnerkonfigurationen installiert und betrieben wird und führt in die Bedienung der in GEM eingebundenen Benutzeroberfläche ein. Zudem werden einige der mitgelieferten Beispiele bzw. Anwendungen beschrieben.
Teil 2 führt in die grundlegenden Programmiertechniken ein. Von einfachen Logikprogrammen über die Entwicklung von größeren Programmen und der Arbeit mit komplexen Datenstrukturen schrittweise bis zu prozeduralen Algorithmen und der Anwendung von Prolog als Metasprache.



Die große Welt der MIDI-Daten

Hardcover
über 430 Seiten
inkl. Diskette
Best.-Nr. B-421
DM 69,- (unverbl. empf. Verkaufspreis)

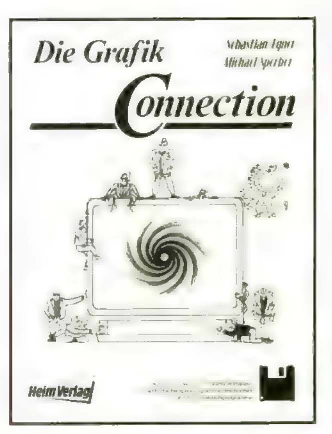
Das Buch erleichtert Anfängern den Einstieg in die Materie und stellt für Profis einen kompetenten Ratgeber und ein vielseitiges Nachschlagewerk dar. Von der Koppelung zweier Synthesizer bis hin zum MIDIfizierten Studio mit Sequenzer, Synchronizer, Computer, Fader-Automation und MIDI-Prozessoren wird ein Einblick in die Möglichkeiten moderner MIDI-Geräte gegeben. Um die Vielseitigkeit des MIDI-Standards nutzen zu können, geht das Buch ausführlich auf die MIDI-Befehle und deren Bedeutung ein. Dem Buch liegt ein komfortabler MIDI-Monitor + Universal-Dump-Utility bei. Dieses MIDI-Scan ist ebenso ein Werkzeug zur Analyse des MIDI-Setups, zur Speicherung von Sysex-Daten wie zum Aufspüren der Geheimnisse, die in der großen MIDI-Welt auf uns warten. Ein umfangreiches Keyboard-Lexikon dient dem Leser als Nachschlagewerk. Die Möglichkeiten moderner MIDI-Geräte und die besten Programme für den ATARI ST werden vorgestellt.



EDV im Büro Organisation und Konzeption

Hardcover
Best.-Nr. B-443
ISBN 3-923250-86-X
DM 49,- (unverbl. empf. Verkaufspreis)

Das Buch geht auf die organisatorische und konzeptionelle Typisierung von EDV-Anwendungen und -Anwendern im Büro ein. Dadurch wird weitgehende Unabhängigkeit von laufenden Soft- und Hardware-Erwägungen erreicht und aus rein organisatorischer Sicht der Schwerpunkt auf den Kern der Büro-EDV gelegt, nämlich auf Kommunikations- und Arbeitsverhalten; Anwendertypen und Anwendungsprofile und die jeweils erforderliche EDV-Unterstützung. Die EDV-organisatorische und konzeptionelle Betrachtung – vom Vorgesetzten bis zur Büro-Hilfskraft – dient der Ermittlung der Software-Typen für optimales Arbeiten. Neu ist auch, daß nicht nur die Vorteile der Büro-EDV dargestellt werden, sondern auch deren Nachteile und schließlich die Risiken einer totalen EDV – für den Anwender und Management. Wer sich ein Bild über organisatorische und konzeptionelle Betrachtungsweisen der Büro-EDV machen will, wird mit diesem Buch genauso zufrieden sein, wie der Praktiker, der sie organisiert.



Die Grafik Connection

Hardcover
über 700 Seiten
Best.-Nr. B-434
ISBN 3-923250-80-0
DM 79,- (unverbl. empf. Verkaufspreis)

Mathematische Computergrafik ist ein attraktives Gebiet der Informatik. Zehn Finalisten des Bundeswettbewerbs „Informatik" haben in der *Grafik-Connection* Grundlagen und Anwendungen zusammengetragen. Sie stellen ein ausgefeiltes Werkzeug für die Programmierung mathematischer Computerprogramme in C vor – das Grafiksystem „Gratia". Es enthält Module für die Programmierung von Punktgrafik, Farbübergängen, geometrischen Formen, Turtle-Grafik und Vektortext. Verzerrungsfreie Darstellung von Bildern ist auf beliebigen Ausgabegeräten möglich, „Gratia" ist mit geringem Aufwand an beliebige Rechner anpaßbar. Auf der Grundlage von „Gratia" wird Computergrafik präsentiert: • Rosetten und Spiralen • Rekursive Kurven und Graphen • Landschaften aus dem Computer • Fraktale in der komplexen Ebene • Objektorientiertes Ray-Tracing • Hyperwürfel und Artverwandte. Ein umfangreicher Anhang mit zusätzlichen Informationen und Listings ergänzt das Buch.

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

BESTELL-COUPON

Bitte senden Sie mir:

______	Einführung in Salix Prolog	à 49,- DM	zzgl. Versandkosten DM 6,-
______	EDV im Büro	à 49,- DM	(Ausland DM 10,-)
______	Die Grafik Connection	à 79,- DM	unabhängig von der
______	Die große Welt der MIDI Daten	à 69,- DM	bestellten Stückzahl

Name, Vorname ______________________
Straße, Hausnr. ______________________
PLZ, Ort ______________________
Oder benutzen Sie die eingeheftete Bestellkarte

In der Schweiz:
Data Trade AG
Landstr. 1
CH - 5415 Rieden-Baden



EDEN

Treffpunkt für jedermann?

Immer mehr Dialogsysteme überschwemmen die DFÜ-Welt. Natürlich macht diese Entwicklung auch vor dem Bildschirmtextsystem nicht halt. Wir untersuchten Eden, das beliebteste BTX-System, auf seine Tauglichkeit.

Stadtplan

Pub 10
Café 11
Bar d'amour ... 12
Post 13
Zeitung 14
Forum 15
Party 16
Clubs 17
Einwohnermeldeamt ... 18
Einwohnerliste 19
Rundbrief 20
Kummerkasten 21

Eden verlassen 99 | weitere Funktionen
33033108A

Bild 1: Der "Stadtplan" von Eden - was machen wir denn heute?

Wer in BTX seinen Spaß haben will, bekommt ihn sehr leicht. Mit *33033# erreicht man Eden, "die phantastische Stadt". Die Übergabeseite verspricht, hier sei ein "Treffpunkt für phantasievolle Menschen". Was auf den ersten Blick zu sehen ist, stellt zwar phantasievolle Pseudonyme dar, die meisten von ihnen jedoch mit Namen, die in eine eindeutige Richtung weisen. Hier erkennt man die "Lust auf Lust", wie es die Aufsichten des Systems nennen, oder einfacher ausgedrückt: Sexbesessene. Man kann nämlich sein Pseudonym frei wählen und wird so von niemandem erkannt bzw. erkennt niemanden. Was macht das System also so interessant?

Nach dem Login in Eden kann man das oben erwähnte Pseudonym definieren. Als nächstes soll man versuchen, seine Person auf einer Visitenkarte zu beschreiben, die alle anderen Teilnehmer abrufen können. Dazu stehen nur 4 Zeilen à 40 Zeichen zur Verfügung - ein sinnloses Unterfangen, wie ich denke. Der Login in das System kostet DM 0,40.

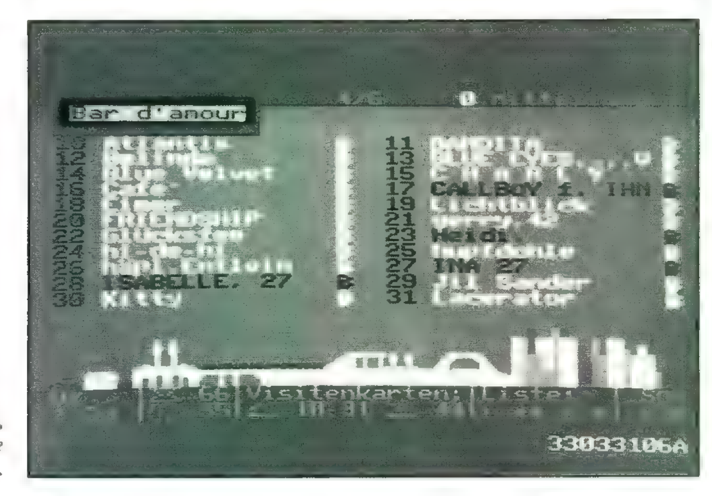


Bild 2: Hupfdohle, Lacerator, Callboy - eine illustre Gesellschaft...

Im System...

...angelangt, sind verschiedene Auswahlen zu treffen - Eden soll ja schließlich eine Stadt darstellen. Wir entscheiden uns für die "Bar d'Amour". Hier angelangt, trifft man zu jeder Tages- und Nachtzeit auf Teilnehmer, die Lust auf Lust haben, also schlicht sexbesessen sind. Dabei sind auch viele Pärchen und Homosexuelle anzutreffen. Denaturierte Lustmolche, sadistische Phrasen, schwülstige Klischees und deklamatorisches Wortgeklapper wirken eher abstumpfend als anregend. Nur zwei Beispiele: "Pärchen sucht jungen Mann zur Besamung" oder "zahle 500,- für IHN..." sind einfach ekelhaft. Deshalb sollten diejenigen, die normale Gespräche vorziehen, sich schnellstens in den "Pub" begeben.

Der Pub hat die gleichen Möglichkeiten wie die Bar d'Amour, allerdings trifft man hier noch auf halbwegs vernünftige Menschen, die einfach nur plaudern wollen. Das System bietet die Möglichkeit, Nachrichten zu schreiben (DM 0,15) und zu empfangen oder Visitenkarten anzusehen (DM 0,15). Wer sich mit diesen Möglichkeiten nicht zufriedengibt, kann auch ein Rundschreiben an die nächsten 50 Benutzer schicken, die nach Eden kommen (DM 7,50). Im Pub kann unter Umständen sogar ein interessantes Gespräch zustandekommen. Positiv: Hier existiert ein harter Kern von Teilnehmern, die vernünftige Diskussionen führen oder nach ihrem Feierabend einfach nur ausgelassen sein wollen.

Natürlich geht es farbig zu. Wie auch im echten Leben existiert in Eden eine "Rassentrennung". Die "schwarzen" Benutzer sind nicht registriert und können deshalb auch keine private Post erhalten. Eine Position weiter folgen in der Rangliste "weiße" Anwender. Diese sind für

einen Monat registriert (DM 2,00) und erhalten ein Paßwort, damit niemand ihren Namen benutzen kann. Dadurch werden sie auch in die Lage versetzt, persönliche Post zu versenden (DM 0,30), die nur der Empfänger der Nachricht selbst lesen kann. Eine noch höhere Priorität besitzt der "gelbe Bürgermeister". Er bzw. sie wird einmal im Monat von den "Einwohnern" Edens gewählt. Alle Kosten, die ihm bei der Ausführung seines "Amtes" entstehen, werden durch die Betreibergesellschaft ersetzt. Den höchsten Rang besitzen die "gelben Aufsichten": Sie können sogar Benutzer aus dem System komplimentieren.

Stadt oder Geldschlucker?

Die Möglichkeiten sind noch nicht ausgeschöpft. Man kann weiterhin wählen zwischen Party (mehrere Benutzer sprechen gleichzeitig miteinander), Café (zwei Benutzer führen einen direkten Dialog), Foren (Diskussionsrunden, in denen die Nachrichten gespeichert werden), Tageszeitung (für Inserate), Gewinnspielen, "Einwohnermeldeamt" usw.

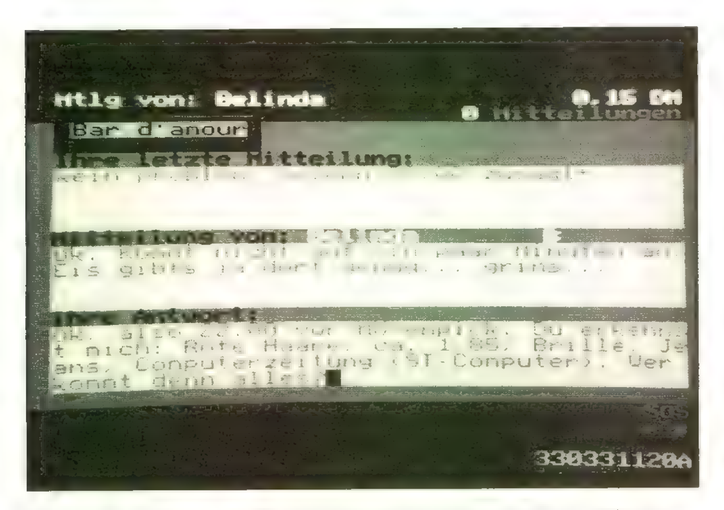


Bild 3: Ein schnelles Treffen mit Belinda im Mövenpick für 15 Pfennig

Eden nennt sich "Stadt". Allerdings sollte man vorsichtig sein, sich allzu oft hier aufzuhalten. Innerhalb von 12 Monaten habe ich für meine Recherchen über 500,- DM ausgegeben, die sich aus "Klekkerbeträgen" von jeweils 15 Pfennigen zusammensetzen - man verliert schnell die Übersicht über das, was man wirklich bezahlt hat. Eden ersetzt auf keinen Fall einen "echten" Dialogpartner und erst recht keine zwischenmenschliche Beziehung. Das scheint jedoch einigen Teilnehmern nicht klar zu sein. Wer nicht hinter dem Monitor verstauben will, sollte lieber einen richtigen Pub aufsuchen.

Fazit

Phantasievolle Menschen findet man bestimmt in Eden. Allerdings nimmt die Phantasie hier manchmal einen merkwürdigen Lauf. Wer einen kurzweiligen Spaß haben möchte, ist mit dem System gut beraten. Dabei darf er/sie sich aber nicht von sexbesessenen Teilnehmern stören lassen und muß unbedingt einen genauen Blick auf seine Ausgaben werfen! Ansonsten: Einfach mal ausprobieren!

MP



Das Jugendzentrum mit viel Grün und viel Technik

Technik für die Jugend

ST im "JuZ 205"

Mitten im schönen Sauerland, in einem verschlafenen kleinen Städtchen namens Hemer (gleich neben Dortmund), befindet sich ein Jugendzentrum, das einige Überraschungen für den computerinteressierten Besucher bietet. Wer würde hier schon Computer oder gar STs vermuten?

Wolfgang Bode, Leiter des Jugendzentrums "205" in Hemer, ist stolz auf die Ausstattung der Freizeitstätte. Seit Mitte 1988 beschäftigen sich einige Mitarbeiter auch im Bereich Computer und haben einiges auf die Beine gestellt. Im Juli '88 trafen sich erstmalig einige Enthusiasten, um eine "AG Computer" ins Leben zu rufen. Wolfgang Bode: "Damals hatten wir nur einen C64 zur Verfügung, das Geld für solche Geräte war im Haushaltsplan nicht vorhanden." Das Gerät gehörte ihm selbst, von diesem Zeitpunkt an stand es allerdings fortwährend im "205". "Stadtverwaltungen sind ungeheuer schwerfällig", fügt er hinzu, "an neue Computer war auf keinen Fall zu denken".

So arbeitete die AG eine Zeitlang mit eigenen, privaten Geräten. Die Zeit wurde größtenteils mit Datenfernübertragung genutzt, doch auch eigene Programme und Spiele kamen nicht zu kurz. Nach einigen Monaten traf dann eine großzügige Spende ein, von der ein komplettes C64-System gekauft wurde (Ende 1988). In Windeseile wurde ein eigenes Mailbox-Programm geschrieben, und es entstand die erste Mailbox in einem Jugendzentrum, die auch rasch Anklang fand.

Vom C64 zum ST

Schnell wurde der alte "Brotkasten" zu klein. "Endlich hatten wir auch Geldmittel von der Stadt bewilligt bekommen", meint Wolfgang Bode, "und wir konnten uns einen Atari ST kaufen. Allerdings waren die Mittel so knapp bemessen, daß wir mit dem Händler einen Sonderpreis aushandeln mußten!" Nachdem das Gerät inklusive Festplatte installiert war, ging die Arbeit erst richtig los. Wieder wurde ein eigenes Mailbox-Programm geschrieben, das bis heute erfolgreich seinen Dienst leistet. Natürlich wurde es wesentlich erweitert. Neben dem üblichen Angebot bietet es ein großes Mailbox-Netz (das MagicNET), ist mit anderen Jugendvereinen und Interessierten lokal vernetzt (über das Aquarius-Netz). Darüber hinaus bietet die Mailbox einen besonderen Service: Die Jugendseite "Szene" der örtlichen Tageszeitung ist immer abrufbar.

Der engagierte Leiter weiß natürlich auch von den Poblemen zu berichten, die er mit dem Aufbau der Mailbox hatte. "Zuerst haben wir mit einem Akustikkoppler gearbeitet. Das größte Problem dabei war, der Post klarzumachen, daß lediglich ein Anschluß benötigt wird, der nur angerufen werden kann. Sobald bürokratische Vorschriften einzuhalten sind, wird die Arbeit für ein Jugendzentrum quälend langsam. Jetzt läuft die Mailbox natürlich mit einem Modem, und wir haben keinerlei Probleme mehr. Außerdem ist der ST absolut geeignet für diese Aufgabe!"

Natürlich wurden auch andere Aktionen ins Leben gerufen. So starteten einige Freaks Kurse für Jugendliche und Erwachsene, die großen Erfolg hatten. Dazu Bode: "Wir hatten mit vielen Problemen zu kämpfen, Kurse für Erwachsene in einem Jugendzentrum abzuhalten. Doch letztendlich kann ein Jugendzentrum auch als Treffpunkt für Jung und Alt dienen, was der Erfolg der Kurse auch eindeutig bewiesen hat."

Nach einigen Spendenaktionen wurde der Gerätebestand noch weiter aufgestockt, immer mehr Geräte und Peripherie kamen nach und nach in den Besitz der Stadt. Was wird heute, nach zwei Jahren, konkret mit den Geräten gemacht? Das kann der Leiter des "205" sehr leicht erklären. "Seit der ST da ist, haben wir vieles auf den Computer ausgelagert. Flugblätter, Programmhefte und Briefe werden nur noch mit 'Signum!' bzw. neuerdings mit 'Script' geschrieben. Dann läuft natürlich immer noch die Mailbox, die im Umkreis einen recht großen Anklang findet. Bei Interesse haben wir natürlich auch preiswerte Kurse für alle Altersgruppen, die regelmäßig gut besucht sind. Neuerdings verlegen wir auch die Kassenabrechnung auf den ST, sind dabei allerdings noch in der Anfangsphase."

"Zum Schluß haben wir sogar mit einigen Interessierten ein unabhängiges Spiele-Testinstitut gegründet, das Computer-Spiele auf ihren pädagogischen Wert hin geprüft hat. Die Aktion ist jedoch im Sande verlaufen, weil ein Großteil der Beteiligten inzwischen in andere Städte

Alle Leser können rechnen, TOS – Leser sparen auch

Adimens 3.0 Plus	101,– DM
Geerdes Softworkstation	35,– DM
Buchhaltung TiM 1.0	49,– DM
GFA-Basic 3.5	40,– DM
GEM-Utility-Package	50,– DM
Easybase	30,– DM
Gesamt	305,– DM



gezogen ist", so Bode. Man sieht also, daß die Ideenküche der "205"-ler immer noch brodelt und immer wieder ungewöhnliche Projekte entstehen, für die ein einzelner gar nicht in der Lage wäre.

Immer auf der Suche

"Geld von einer Stadtverwaltung zu erhalten, ist immer schwierig." So sucht Wolfgang Bode immer Spenden aller Art. Doch nicht nur Geräte werden gesucht, auch Interessierte sollen sich bei ihm melden. "Bei uns kann jeder seine Wünsche verwirklichen, hier geht es nicht so starr zu, wie man es vielleicht vermutet", betont er. "Die Räumlichkeiten und Geräte sind vorhanden, und wer Ideen hat, ist bei uns an der richtigen Stelle."

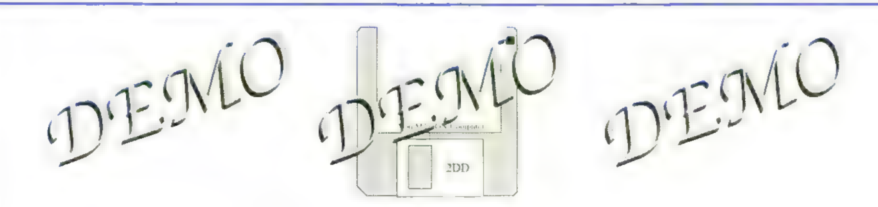
Für Computer-Fans ebenfalls interessant sein dürfte die Rollenspielgruppe. Hier werden jeden Freitag bis spät in die Nacht Orks und Drachen geprügelt - dabei wird es selten langweilig! Und obwohl man AD&D-Gruppen selten genug findet, treffen sich hier jeden Freitag mindestens fünf Interessierte, die ebenfalls immer neue Mitspieler suchen. Weiterhin im Angebot ist ein regelmäßiges Kinoprogramm für einen geringen Eintritt (um 5,- DM). Der Freak, der sich vom Computer erholen will, kann auch die regelmäßig stattfindende Disco besuchen (mit großer Sound- und Lichtanlage), Billard spielen (ein eigener Verein existiert), einfach nur klönen, Leute kennenlernen usw.

Wer jetzt Interesse hat, sollte sich schleunigst melden. Entweder direkt:

Jugendzentrum "205"
Hauptstraße 205
5870 Hemer
Tel. (02372) 73025

oder aber in der Mailbox unter der Telefonnummer (02372) 13383. Die Parameter: 300, 1200 und 2400 Baud, 8 Daten-Bits, 1 Stopbit, keine Parität, 24 Stunden online - natürlich kostenlos.

MP



Demo-Disketten

Damit Sie nicht immer die Katze im Sack kaufen müssen, haben wir ab sofort eine neue Rubrik für Sie eingeführt; es sind Demo-Disketten kommerzieller Software. Sie kosten lediglich DM 10,- pro Diskette und können über die Redaktion bezogen werden. So müssen Sie zum Vergleich verschiedener Programme nicht an verschiedene Hersteller schreiben, sondern können sich in aller Ruhe das Demonstrationsprogramm ansehen, bevor Sie das Original kaufen.

Bitte beachten Sie, daß die angebotenen Disketten nur Demonstrationsdisketten der Originalversionen sind und somit im Gegensatz zu den Originalen in Funktion eingeschränkt sind!

Folgende Demo-Disketten sind z.Zt. erhältlich:

D1: S.&P.-Charts
Chart-Analyseprogramm
(*S.P.S. Software*)

D2: SPC-Modula-2
Modula-2-Entwicklungssystem
(*Advanced Applications Viczena*)

D3: ST-Fibu
Finanzbuchhaltungsprogramm
(*GMa-Soft*)

D4: ST-Fibu-Fakt
Fakturierungsprogramm für ST-Fibu
(*GMa-Soft*)

D5: ST-Fibu-Text
Textverarbeitungsprogramm für ST-Fibu mit Serienbrieffunktion
(*GMa-Soft*)

D6: SciGraph
Programm zur Erstellung von Präsentationsgrafiken
(*SciLab GmbH*)

D7: ST-Statistik
Uni- und multivariates Statistikprogramm, Grafikeinbindung (*SciLab GmbH*)

D8: fibuSTAT
Finanzbuchhaltungs-/Statistikprogramm
(*novoPLAN Software GmbH*)

D9: Btx/Vtx-Manager
Programm zum Anschluß an Bildschirmtext
(*Drews Btx + EDV GmbH*)

D10: Edison
Editor für fast alle Gelegenheiten
(*Kniss Soft*)

D11 & D12: CADjA
CAD-Programm für hohe Ansprüche
(*Computer Technik Kieckbusch*).
Demo besteht aus zwei Disketten zu je DM 10,-!

D13: JAMES 2.0
Programm für Börsenspekulanten
(*IFA-Köln*)

D14: Soundmerlin
Sample-Editor-Programm mit vielen Modulen
(*TommySoftware*)

D15: Soundmachine II
Programm zur Erstellung und Wiedergabe von Sounds
(*TommySoftware*)

D16: ReProK
Büroorganisationsprogramm
(*Stage Microsystems*)

D17: Sherlook
Schrifterkennungs- und -verarbeitungsprogramm
(*H.Richter*)

D18: ST Matlab
Programmiersystem mit Schnittstelle zu Modula-2
(*Advanced Aplications Viczena*)

D19: Calamus
Desktop-Publishing-Programm
(*DMC*)

D20: GD-Fibu
Finanzbuchhaltungsprogramm
(*GDAT*)

D21: Omikron.Draw!
Zeichen- und Malprogramm
(*Omikron.Software*)

D22: Omikron.Libraries
Verschiedene Libraries für Omikron.BASIC
(*Omikron.Software*)

D23: Omikron.Compiler
Demo-Version des Omikron.BASIC-Compilers
(*Omikron.Software*)

D24: Mortimer
Multi-Programm für alle Gelegenheiten
(*Omikron.Software*)

D25: Script
Textverarbeitungsprogramm
(*Application Systems /// Heidelberg*)

D26: SuperScore
Sequencer- und Notendruckprogramm
(*BELA Computer GmbH*)

D27: SPS-Emulator
Programm zur Simulation von SPS-Steuerungen
(*Karstein Datentechnik*)

D28: STAD 1.3+
Zeichenprogramm mit 3D-Teil
(*Application Systems /// Heidelberg*)

D29: MegaFakt
Fakturierungsprogramm
(*MegaTeam*)

D30 & D31: MegaPaint II
Zeichenprogramm mit Vektorteil
(*TommySoftware*)

D32: Tempus Word
Textverarbeitung
(*CCD*)

D33: Creator
Zeichenprogramm mit Animationsteil
(*Application Systems /// Heidelberg*)

D34: Outline Art
Utility für Calamus
(*DMC*)

D35: compugraphic Schriften
für Calamus
(*DMC*)

D36: BTX-Börsen-Manager
Börsenprogramm
(*Thomas Bopp Softwarevertrieb*)

D37: Cashflow
Kassenbuch
(*C.A.$.H.*)

D38: TiM II
Finanzbuchhaltungsprogramm
(*C.A.$.H.*)

D39: Merkator
(Finanzbuchhaltungsprogramm)
(*MDC*)

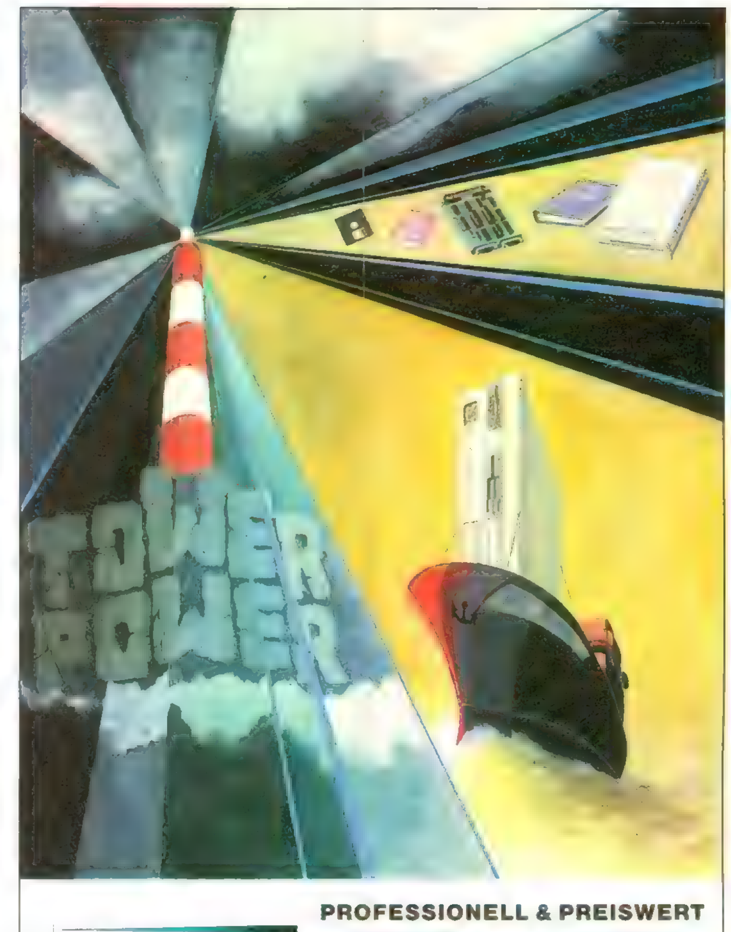
D40: Technobox Drafter
(Zeichenprogramm spez. f. Konstruktionen)
(*Technobox*)

D41: Platon
(Leiterplatten- CAD-System)
(*VHF-Computer*)

Es gelten die gleichen Vertriebsbedingungen wie für PD-Disketten (s. PD-Seiten am Ende dieser Ausgabe). Demo-Disketten können auch zusammen mit PD- und Sonder-Disketten bestellt werden.

Bitte vergessen Sie nicht die betreffende Bestellnummer (z.B. D1) anzugeben.







KOSTENLOSEN KATALOG ANFORDERN

PROFESSIONELL & PREISWERT

ZUBEHÖR und SOFTWARE für Ihren ATARI* im BÜRO oder ZUHAUSE

* ATARI ist ein eingetragenes Warenzeichen der Atari-Computer GmbH

Riedstr. 2 – 7100 Heilbronn – Tel. 07131/78480

Lektorat

Die Rechtschreibkorrektur

liest SIGNUM (2)!, 1st Word+, ASCII (TeX u.a.)
110 000 Wörter im Standard-lexikon – sehr schnell, Korrektur mit bis zu 15 Lexika, Textstatistik, Deklination, Konjugation, sichere Trennung (99%), Groß-, Klein-schreibung u.v.m.

149,–

MS–DOS

Super Charger
incl. MS-DOS 4.01 648,–

VORTEX ATonce
AT Emulator 425,–

AT Speed a.A.

MAXON MGE II mit EIZO 6500
in unserem DTP Center vorführbereit

Festplatten VORTEX Datajet

SCSI – 25 ms – kaum hörbar

30 MB	1088,–
40 MB	1388,–
60 MB	1648,–
90 MB	2288,–
130/180 MB	a.A.

44 MB Wechselplatte
incl. Medium 2288,–
Wechselpl. 44 MB mit Festpl.

+40 MB	3488,–
+90 MB	4288,–
+ 130/180 MB	a.A.

aktuelle

That's Write	285,–
Word Perfect	685,–
Signum2!	a.A.
Calamus	675,–
Outline Art	335,–
Arabesque	210,–
STAD 1.3+	a.A.

fibu man 4.0, T.I.M. Iu.II

Software

Adimens 3.0 +	295,–
Adimens 2.3	135,–
Aditalk 3.0 +	295,–
Themadat 4.0	205,–
Cadja	895,–
Turbo ST 1.8	85,–
Megapaint II prof.	695,–

und weitere Softw. a.A.

Prof. Layout für Calamus

1) Schnitt- & Passmarken DIN A4, A5, A3 hoch und quer 79,–
2) Geschäftspapiere 99,–
3) Familiendrucksachen 79,–
4) Werbe-Design für Prospekte, Folder, Anzeigen etc. 99,–

Calamus Belichtung
6301 Reiskirchen – Grünbergerstr. 31
Tel. 0 64 08 / 6 33 34 Fax 6 29 75

Großbildschirme, Scanner, usw. a.A.

SCSI Speed Drive Festplatten
(Hard & Soft) a.A.

MEGA ST 2	1888,–
NEC P6+	1188,–
Cartridge 44 MB	235,–
Logitech Maus	99,–

Auf Wunsch Festplatten mit PD-Software (MAXON ... 346) 1 MB ... DM

Festplatten VORTEX HD plus

20 MB	848,–
30 MB	948,–
60 MB	1198,–
120 MB	1798,–

Platten- und Lüfterabschaltung !!!
Autopark, Autoboot, Cache u.v.m., erstkl. Software

PROTAR profile

Festplatten

SCSI – sehr leise – 1–2 J. Gar.

30 MB, 40 ms	998,–
40 MB, 28 ms	1298,–
40 MB, 19 ms	1398,–
60 MB, 28 ms	1598,–

44 MB Wechselplatte
incl. Medium 1998,–

20, 80 u. 160 MB a.A.

WAVE

Computersysteme & Softwaredistribution

Südanlage 20 6300 Gießen
TEL: 0641 – 72357 FAX: 72371

NEU NEU NEU NEU NEU NEU NEU NEU NEU NEU NEU NEU NEU NEU NEU

KOLIBRI GRAFIK GMBH
ATELIER FÜR GRAPHIK & BESCHRIFTUNGSTECHNIK
45 OSNABRÜCK MÖSERSTR. 2
TEL 0541/22422

Wir schneiden Ihre Vektorschriften & Graphiken in CVG, GEM u. VEK-Format, aus

SELBSTKLEBEFOLIEN

Für den Privatgebrauch & für Beschriftungen von Schildern, Fahrzeugen u. Lichtwerbeanlagen

CUT–service

sprechen Sie mit uns !

Was Sie schon immer über

ELEKTRONISCHE BILDVERARBEITUNG

wissen wollten,
aber bisher nie zu fragen wagten

Immer mehr Atarianer entdecken die Welt der Elektronischen Bildverarbeitung (kurz EBV). Inzwischen versteht sich der ST gar prächtig mit hochwertigen Satzbelichtern, die mit Auflösungen aufwarten, von denen man vor ein paar Jahren nur geträumt hat. Das nötige Werkzeug, um ordentliche Bildreproduktionen und -manipulationen vornehmen zu können, ist inzwischen auch vorhanden. Nicht zuletzt durch die Verbreitung von CALAMUS hat die ATARI-Szene enormen Zuwachs an professioneller Software rund um das Thema DTP erhalten, die sich vor jener aus der MAC-Welt nicht zu verstecken braucht. Vielen Einsteigern fehlt allerdings das dringend notwendige Hintergrundwissen. Gerade bei den Auflösungen hochwertiger Belichter sind viele Kunden über die Ergenisse ihrer Scan-Kunst verwundert. Wir wollen mit diesem Artikel versuchen, Ihnen den Einstieg in das Thema "Elektronische Bildverarbeitung" zu erleichtern.

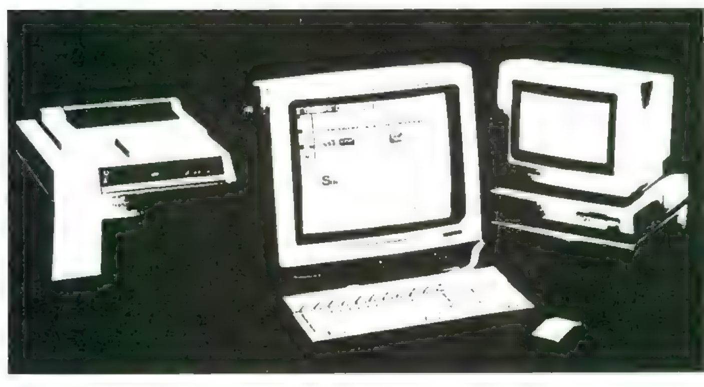


Bild 1: Scannen ist nicht so einfach

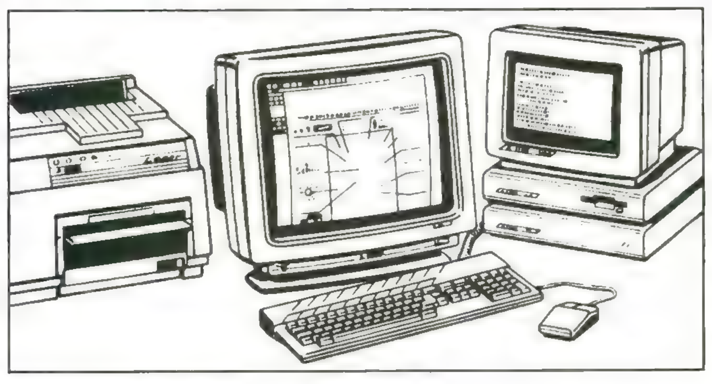


Bild 2: Die Strichvorlage

Strich- und Halbton-Reproduktion

Angenommen, Sie möchten sich bei Ihrer Großmutter für die großzügige Finanzspritze beim Kauf Ihrer neuen Computeranlage bedanken. Sie machen ein Foto von dem guten Stück, legen es unter Ihren neuen Scanner, lesen es in Ihren Computer ein - und wundern sich, weshalb das eingelesene Bild so überhaupt nichts mehr mit der Vorlage gemeinsam hat (Bild 1).

Was ist passiert? Scanner kaputt? Vorlage nicht "computer-reproduzierbar"? Weder noch: Sie müssen dem Scanner mitteilen (sofern er Sie versteht), was er mit dem Bild machen soll. Er muß wissen, ob es sich um eine Strichvorlage oder um eine Halbtonvorlage handelt.

Strichvorlagen

Bei Strichvorlagen liegen alle Bildelemente in nur einer Tonstufe vor. Am besten ist absolute Schwärzung auf weißem Papier. Sobald sich zwischen die beiden extremen Werte (man spricht von "Licht" für den helleren und "Tiefe" für den dunkleren) weitere Graustufen schieben, handelt es sich um eine "Halbtonvorlage". In Ihrem Fall handelt es sich also um eine Halbtonvorlage. Die darin enthaltenen Graustufen muß der Scanner also als solche erkennen können, damit Ihre Reproduktion dem Original ähnlich sieht. Kann er's nicht, dann ordnet er jedem eingelesenen Punkt entweder schwarz oder weiß zu - je nachdem, wie Sie die Helligkeit eingestellt haben. Sie besorgen sich also der Einfachheit halber eine Strichvorlage (meist als "Zeichnung" bezeichnet) und machen damit das Gleiche noch einmal: Der Erfolg ist durchschlagend und vom Original kaum zu unterscheiden (Bild 2).

Normale Strichvorlagen (also auch gedruckter Text) sind also auch für kleinere EBV-Systeme kein größeres Problem. Haariger wird die Angelegenheit bei sogenannten "Feinstrichvorlagen". Dieses sind z.B. sehr fein strukturierte Zeichnungen oder Bilder mit einer sehr feinen Schraffur. Nehmen wir an, eine technische Zeichnung besitzt eine Schraffur von sagen wir 40 Linien pro cm (l/cm). Dies entspricht einer Liniendichte von ungefähr 100 Linien pro Inch (lpi). Ihr Scanner hat eine Auflösung von 200 dots per inch (dpi), d.h. er besitzt 200 Sensoren pro Inch. Er wird also Schwierigkeiten haben, diese Schraffur noch sauber zu trennen, denn dazu müßten die Linien bzw. die Linien-Zwischenräume genau unter einem Sensor des Scanners und noch dazu parallel zur Abtastrichtung liegen. Man

sagt, daß man die besten Ergebnisse bei Strichvorlagen erhält, wenn die "maximale Auflösung" der Vorlage nicht mehr als 1/4 der Scanner-Auflösung beträgt. So sind immer zwei Sensoren für eine Linie bzw. Linien-Zwischenraum zuständig.

An dieser Stelle möchte ich Sie auf eine sehr unschöne Gepflogenheit der Scanner-Hersteller aufmerksam machen. In den Werbebroschüren taucht sehr selten das Wort "Auflösung" auf. Vielmehr wird von "Ausgabeauflösung" gesprochen. Der Grund dafür ist folgender: Die meisten Scanner, die angeblich 400 oder 600 dpi "schaffen" sollen, arbeiten nur mit der Hälfte der Resolution und interpolieren auf die gewünschte "Ausgabeauflösung". Das heißt, der Scanner hat z.B. "nur" 300 Sensoren pro Inch und vergrößert das Bild intern um den Faktor zwei. Sie können dies leicht anhand des Datenblattes Ihres Scanners überprüfen: 2552 Sensoren pro 21.6 cm entsprechen einer Sensorendichte von 300 Stück pro Inch - also beträgt die maximale Auflösung des Scanners 300 dpi - mehr nicht.

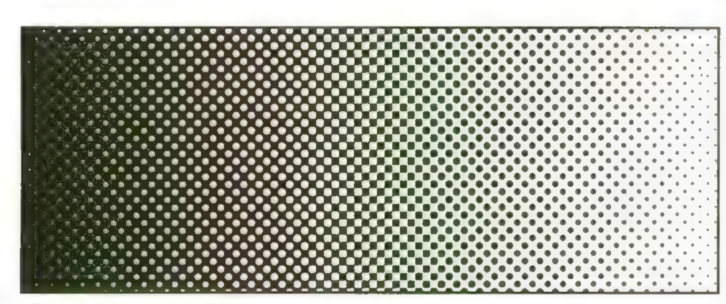


Bild 3: Rasterverlauf

Halbtonvorlagen

Wenden wir uns wieder dem Bild für Ihre Oma zu. Nachdem Sie jetzt wissen, worauf es bei Strichvorlagen ankommt, kramen Sie sich wieder das Foto hervor und versuchen Ihr Glück noch einmal. Der Scanner muß die Graustufen des Bildes erkennen. Die Fläche des Fotos wird horizontal und vertikal in eine bestimmte Anzahl von Punkten aufgeteilt. Es wird digitalisiert. Jedem dieser Punkte ist ein bestimmter Grauwert oder, anders gesagt, eine bestimmte Helligkeit zugeordnet. Es hängt jetzt davon ab, wieviele Graustufen Ihr Scanner unterscheiden kann. Heute arbeiten EBV-Systeme auf der Basis von 256 Graustufen. Das ist eine ausreichende Anzahl, um Fotoqualität zu erreichen. Man sagt, der Scanner liefert die Bildpunkte in "acht Bit Tiefe". Sie wissen, daß man mit acht Bit 256 (2^8=256) verschiedene Werte unterscheiden kann. Dazu ein wichtiger Merksatz vorab: Bildpunkte, die bei der Digitalisierung entstehen, und Rasterpunkte, die für das Drucken benötigt werden, sind zwei verschiedene Paar Stiefel: Die Bildpunkte stehen am Anfang des gesamten Bildverarbeitungsprozesses, die Rasterpunkte am Ende. Sie repräsentieren die Endqualität.

Gänzlich ungeeignet für die EBV sind Scanner, die zwar über einen Graustufenmodus verfügen, aber die einzelnen Grauwerte nicht preisgeben, indem sie die Vorlage intern aufrastern, was eigentlich das EBV-Programm bei der Ausgabe tun sollte. Das bedeutet, daß der angeschlossene Computer nicht mehr erkennen kann, wie die eigentliche Helligkeit des Ur-Bildpunktes gewesen ist, denn er erhält die Bildinformation bereits umgesetzt in ein reines Schwarzweißraster. Man kann es nicht mehr für EBV-Programme verwenden, da diese unbedingt die Helligkeit eines jeden Bildpunktes benötigen. Wenn Sie also gute Halbtonreproduktionen erstellen wollen, achten Sie unbedingt darauf, daß der Scanner die Bildpunkte in einer Abstufung von acht Bit bereitstellt. Das Prädikat "256 Graustufen" ist also aus dem oben beschriebenen Grund nicht unbedingt ausreichend.

Jetzt können Sie sich auch den Unterschied zwischen .IMG-, .PI3-, .PIC- oder .PAC-Bildern und .TIF-, .RTC-, oder .ESM-Bildern erklären. Erstere sind Bilder, die nur entweder weiße oder schwarze Bildpunkte (ein Bit Tiefe) enthalten, letztere dagegen "echte" Halbtonbilder mit 256 Graustufen in "acht Bit Tiefe". Sie benötigen deswegen auch achtmal mehr Speicher.

Sie legen also Ihr Foto wieder in den Scanner, stellen ihn auf "Halbtonmodus", und los geht's - vorausgesetzt, Sie haben die geeignete Software, um Halbtonbilder verarbeiten zu können. Wenn nicht, dann hätte Ihnen ein intern rasternder Scanner genügt.

Bevor wir uns die Möglichkeiten der Bildmanipulation bei EBV-Systemen ansehen, möchte ich erst das Thema Bildausgabe behandeln.

Drucken oder drucken lassen - das ist hier die Frage

Um die vielen Graustufen der Fotos im Druck sichtbar zu machen, muß man ein spezielles Verfahren anwenden, da kein Drucker "echte" Graustufen erzeugen kann. Ein Drucker (sei es Nadel-, Laser-, Thermodrucker oder Satzbelichter) kann nur schwarze Punkte darstellen. Damit dennoch der der Eindruck von Grauabstufungen entseht, wendet man einen Trick an, der das menschliche Auge überlistet: Man ordnet die kleinen,schwarzen Punkte auf dem Papier so an, daß sie vom Auge als solche nicht mehr erkennbar sind. Aus dem Anteil von schwarzen Rasterpunkten an dem weißen Papier in einer bestimmten Bildfläche ergibt sich für das Auge ein bestimmter Grauwert. Das ist das Prinzip des Rasterns. Abstufungen zwischen Vollton (Schwarz) und Weiß werden durch verschieden große Rasterelemente vorgetäuscht, die eigentlich Strichcharakter haben und deswegen auch "unechte Halbtöne" genannt werden (Bild 3).

Die Rasterweite wird in Linienzahl pro cm angegeben. Ein 60er Raster besitzt also 60 Rasterlinien pro cm. Dabei liegt es auf der Hand, daß die Druckerauflösung damit in unmittelbarem Zusammenhang steht. Mit einem 300-dpi-Laserdrucker (300 dpi entsprechen knapp 120 Linien/cm) können Sie kein 60er Raster (60 Linien/cm) bei 256 Graustufen drucken, denn bei dieser Auflösung können Sie die Größe der Bildpunkte nicht mehr modulieren.

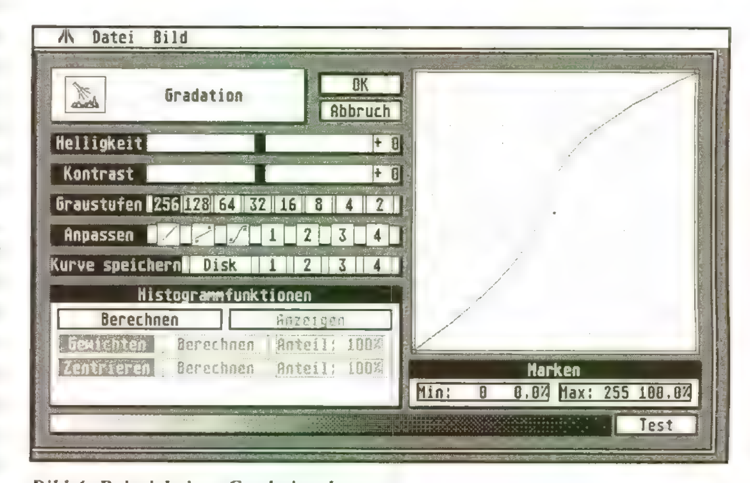
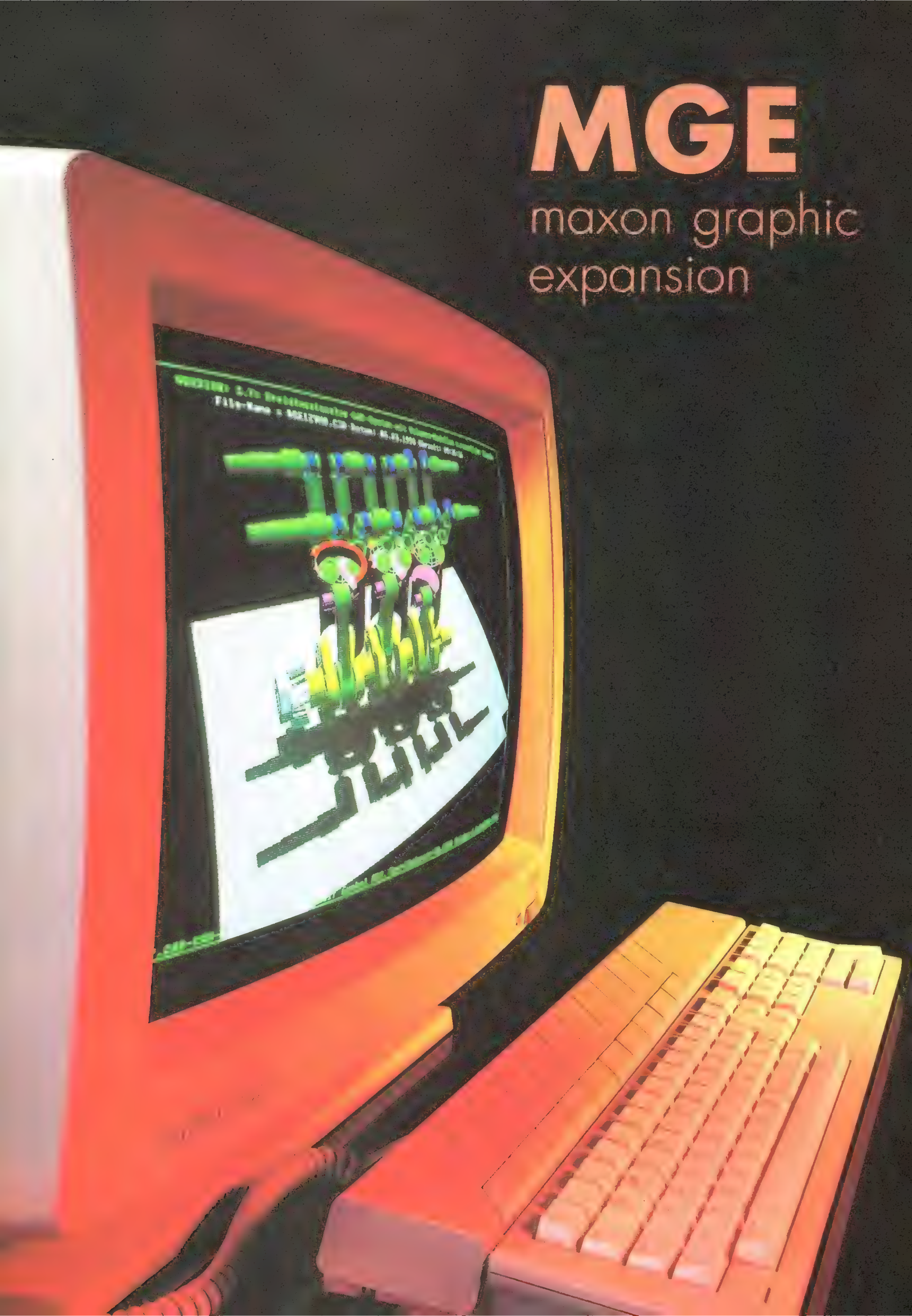


Bild 4: Beispiel einer Gradationskurve

Aus diesem Grund werden die gerasterten Fotos meist auf Satzbelichtern ausgegeben, die mit einer Auflösung von mehr als 2500 dpi arbeiten. Hier kann man leicht eine Rasterweite von 60 L/cm oder mehr einstellen: das Raster ist nur noch mit der Lupe auszumachen, die Auszüge (man


MGE
maxon graphic
expansion

Die Grafikerweiterung für Profis

Vorbei sind die Tage der Eintönigkeit, die MAXON Graphic Expansion bringt Farbe ins Spiel: Gleichzeitig bis zu 256 Farben oder echte Graustufen. Auflösungen von bis zu 1664*1200 Bildpunkten, blitzschneller Bildaufbau durch einen leistungsfähigen Grafikprozessor - dies sind die Merkmale, die die MGE zur idealen Grafikerweiterung für Profis machen. Speziell ausgerichtet auf die Anforderungen kreativer Anwendungen wie DTP, CAD oder elektronischer Bildverarbeitung eröffnet sie durch ihre enorme Flexibilität neue Dimensionen der Produktivität und verwandelt den Mega ST in eine Grafik-Workstation.

Intelligenz und Stärke:

INTELs Grafikprozessor 82786 treibt die MGE zu grafischen Höchstleistungen. Die serienmäßige CLUT stellt eine Palette von 16,7 Millionen Farben zur Verfügung. Die flexible Programmierbarkeit erlaubt beliebige Bildwechselfrequenzen. Ein freier Sockel lädt zum Einsatz eines mathematischen Coprozessors ein. Die Fähigkeit zur externen Synchronisation ermöglicht den Einsatz der MGE in Video- oder Genlock-Anwendungen. Und in ihrem Drang, bis zu 2 Millionen Bildpunkte gleichzeitig darzustellen, wird die MGE meist nur vom angeschlossenen Monitor zurückgehalten.

Einsetzen und loslegen:

Durch die GDOS-VDI-Treiber sind GEM-Programme auf der MGE lauffähig. Für nonkonforme Software wird eine S/W-Großbildschirmemulation mitgeliefert. VDI-Druckertreiber und Monitorbibliotheken sorgen sich um Routinejobs. Das Kontrollfeld ermöglicht die freie Einstellung von Auflösung und Farben. Und mit den mitgelieferten Bindings für Turbo-C, Assembler und GFA-Basic wird jeder Programmierer zum Herrn der Formen, Farben und Frequenzen.

Geprüft und für gut befunden:

So urteilt die Fachpresse über die MGE. Etwa c't 5/90: "Die konzeptionelle und praktisch realisierte Vielseitigkeit dieser Karte ... basiert auf den exzellenten Eigenschaften des eingesetzten Grafikprozessors." Ebenso COMPUTER LIVE 8/90: "Für alle Atari-Besitzer, die sich auf professionellem Niveau mit DTP, CAD oder Desktop-Video beschäftigen wollen, bietet MAXON mit der MGE II eine schnelle, extrem vielseitige und leistungsstarke Lösung." Oder TOS 6/90: "Ein Eldorado für alle programmierenden Grafik-Enthusiasten..." Auch PAGE 6/90: "Hier hebt sich positiv die MGE II von MAXON Computer hervor..." Und schließlich ST Magazin 8/90: "Fazit: Grafikkarte für professionelle DTP- und CAD-Anwendungen." Da bleibt eigentlich nur die Frage offen, was die Tester wohl zur neuen Version mit der serienmäßigen Palette von 16,7 Millionen Farben gesagt hätten.

Daten und Features:

Grafikprozessor:	Intel 82768
Bildspeicher:	1024 KByte
Farbpalette:	16,7 Millionen
gleichzeitig darstellbare Farben/Graustufen:	256
Pixelfrequenzen:	13.75, 27.5, 55 und 110 MHz
Betriebsarten:	Interlace/Non-Interlace

Typische Auflösungen:

Interlaced (Halbbildfrequenz):
- 1664 x 1200 Pixel monochrom, 88 Hz
- 1280 x 1024 Pixel in 16 Farben, 65 Hz
- 896 x 684 Pixel in 256 Farben, 66 Hz

Non-Interlaced (Vollbildfrequenz):
- 1280 x 960 Pixel monochrom, 65 Hz
- 896 x 688 Pixel in 16 Farben, 67 Hz
- 640 x 480 Pixel in 256 Farben, 66 Hz

Besonderheiten: Anschlußmöglichkeit für Genlock oder externe Synchronisation, Sockel für mathematischen Coprozessor 68881

Software: Installationsprogramm, Treiber für Hardware, GDOS-VDI und Drucker, Monitorbibliotheken, Kontrollfeld als Accessory, Diverse Demos und Bilder, Programmierbibliotheken

Preise und Systeme:

MGE II	DM 2398,-
MGE II mit Eizo 9070S (16" Farbmonitor)	DM 5498,-
MGE II mit Eizo 6500 (21" Graustufengroßbildschirm)	DM 6298,-
MGE II mit Eizo 9400 (20" Farbgroßbildschirm)	DM 8998,-

Sehen und staunen:

Gerne führen Ihnen die folgenden Händler die MGE vor:

Stenknecht
Bürokommunikation
Heiligengeiststr. 20
2120 Lüneburg
Tel. 04131/46122

MCC
Holzkoppelweg 19a
2300 Kiel 1
Tel. 0431/54381

PS-DATA
Hard- und Software
GmbH
Faulenstr. 48-52
2800 Bremen
Tel. 0421/170577

COM-DATA GmbH
Schiffgraben 19
3000 Hannover 1
Tel. 0511/326736

3K
Computerbild GmbH
Sassenfeld 71
4054 Nettetal 1
Tel. 02153/60001

CSA Computersysteme
Hüttenstr. 56
4650 Gelsenkirchen
Tel. 0209/203420

Data Pach
GmbH & Co KG
Neuhausstr. 4
5100 Aachen
Tel. 0241/4778537

Werbestudio Abakus
Römerstraße 24
5300 Bonn 1
Tel. 0228/635712

Lehr Bürocenter GmbH
Gülerstraße 82
5500 Trier
Tel. 0651/209710

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt 90
Tel. 069/763409

Pauly Büromaschinen
Salzgasse 6
6250 Limburg/Lahn
Tel. 06431/50040

WAVE
Computersysteme
Südanlage 20
6300 Gießen
Tel. 0641/72357

Schreiber Computer
Rotebühlplatz 10
7000 Stuttgart 1
Tel. 0711/221996

Walliser & Co KG
Marktstr. 48
7000 Stuttgart 50
Tel. 0711/567143

Comp & Phone
Alleenstr. 66
7312 Kirchheim-Teck
Tel. 07021/3949

GCS Roseler
Jahnstr. 1
7343 Kuchen
Tel. 07331/81925

Besch & Partner
Stuttgarter Str. 53
7432 Bad Urach
Tel. 07125/8199

Erhardt Bürotechnik
Am Ludwigplatz
7500 Karlsruhe 1
Tel. 0721/16080

Jost Computer
Im Wendelrot 5
7520 Bruchsal
Tel. 07251/103091

Computer Ludwig
City Studio
Am Rindermarkt 6
8000 München 2
Tel. 089/2609801

JOBI'S
Unternehmenssupport
DV & Marketing
Fliegenstr. 12
8000 München 2
Tel. 089/2608703

HIB Computer
DTP-Center
Äußere
Bayreuther 57a-59
8500 Nürnberg 10
Tel. 0911/995140

CCN Computer Center
Nürnberg GmbH
Gibitzenhofstr. 86
8500 Nürnberg 70
Tel. 0911/421056

V. Willgerodt
Bürobedarfshaus
GmbH & Co KG
Hanauer Str. 12
8750 Aschaffenburg
06021/21375

Händleranfragen erwünscht!



MAXON Computer • Schwalbacherstr. 52 • 6236 Eschborn
Tel.: 0 61 96 / 4818 11 • FAX: 06196/41885

spricht auch von Lithografien) erreichen absolute High-End-Qualität.

So ein Belichter kostet mehr als 120.000 DM, also wundern Sie sich nicht, daß die Belichtung einer DIN A4-Seite um die 15 DM kostet. Das Belichten einer Seite von Calamus aus kann, je nachdem wieviele Bilder darauf enthalten sind, bis zu 20 Minuten oder länger dauern. Gerade in der noch jungen Belichterbranche gilt aber der Grundsatz: "Time is money". Also legen Sie Ihrer Diskette/Wechselplatte am besten einen Zettel bei, auf dem genau steht, für welche Auflösung Sie Ihre Bilder optimiert haben. Dies erspart dem Operator einen Anruf bei Ihnen.

Eine zweite (lower-cost-) Möglichkeit besteht darin, das Bild über Laserdrucker im 20er-Raster auszugeben und dann mit Zoom-Kopierer zu verkleinern, wobei die Rasterweite natürlich mit verkleinert wird.

Bilder sind Speicherfresser

An dieser Stelle möchte ich kurz auf den Speicherbedarf von digitalisierten Bildern eingehen. Man stelle sich vor: Jeder einzelne Bildpunkt muß in der Lage sein, 256 verschiedene Helligkeiten zu unterscheiden, das heißt, man benötigt ein Byte pro Bildpunkt. Ausgehend von dem Beispiel des 60er-Rasters wären dies 60x60 Punkte pro cm2, also 3600 Punkte. Das heißt, daß ein einziger Quadratzentimeter 3.6k Speicher benötigt. Will man ein 20x30 cm großes Bild in High-End-Qualität (zwei Bildpunkte pro Rasterpunkt) bearbeiten, werden dafür sage und schreibe 4.3 MByte benötigt. Deshalb arbeiten leistungsfähige EBV-Programme mit virtueller Speicherverwaltung, d.h. es ist immer nur der sichtbare Bereich des Bildes im Hauptspeicher des Rechners, der Rest befindet sich auf der Festplatte und wird bei Bedarf hinzugeladen.

Auflösung ist nicht alles..

Auch die Auflösung, mit der Sie Ihren Scanner betreiben, steht mit dem späteren Ausgaberaster in enger Beziehung. Angenommen, Sie drucken das Foto für Ihre Oma mit einem Laserdrucker aus, dann können Sie maximal mit einem 20er-Raster arbeiten. Sie sollten mindestens einen Bildpunkt pro Rasterpunkt haben, optimale Ergebnisse erreicht man, wenn man zwei Bildpunkte pro Rasterpunkt hat. Also: 20 L/cm sind ungefähr 50 dpi. Das Ganze mal zwei (da wir zwei Bildpunkte pro Rasterpunkt haben) macht 100 dpi. Es sind also keine wesentlichen Qualitätsverbesserungen zu erkennen, wenn Sie das Bild mit 300 dpi einscannen. Nur wenn Sie das Bild vergrößern wollen, ist eine entsprechend höhere Scanner-Auflösung sinnvoll.

Die Software machts...

Nun haben Sie also Ihr Bild endlich als Halbton in Ihren Rechner eingelesen. Als glücklicher Besitzer einer Grafikkarte und eines Graustufen- oder Farb-MultiSync-Monitors haben Sie das Bild in Fotoqualität vor sich. Aber auch diejenigen, die "nur" über den ATARI-Originalmonitor SM 124 verfügen, brauchen nicht im Dunkeln zu tappen - das Bild wird auch auf diesem Monitor dargestellt. Die Graustufen ergeben sich wieder durch eine speziell für den Monitor geeignete Rastermethode, genannt "Ordered Dither". Dabei wird nicht die Punktgröße, sondern die Punktanzahl pro Fläche verändert.

Jetzt können Sie Ihr Bild nach Herzenslust bearbeiten. Was Sie damit anstellen, bleibt Ihnen überlassen. Je nach Programm können Sie Bilder überlagern, Bereiche verzerren, Weichzeichner einblenden usw.

Wenn Sie vorhaben, das Bild über eine Offsetdruckanlage vervielfältigen zu lassen, müssen Sie sich unbedingt über den Tonwertumfang informieren, der mit der Maschine maximal möglich ist. Im Druckerjargon spricht man nicht - wie beim Digitalisieren - von 0 bis 255 Graustufen, sondern von Prozentwerten. Das bedeutet 0% ist kein Punkt, 100% bedeutet schwarze Fläche. Üblicherweise besteht ein Tonwertumfang von 7%-15% im Licht und 80%-90% in der Tiefe. In allen Bereichen, die heller sind als z.B. 9%, kann es passieren, daß beim Druck die "Punkte wegbrechen". Umgekehrt drohen Bereiche, die dunkler sind als z.B. 88%, beim Druck "zuzuschmieren". Deswegen können Sie bei EBV-Programmen den Tonwertumfang auf das benötigte Maß reduzieren. Dies geschieht in den meisten Fällen über die Gradationskurve.

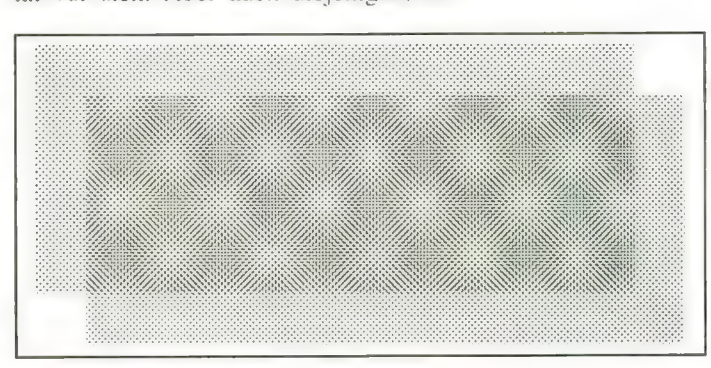


Bild 5: Ein Moiré kann entstehen, wenn sich zwei Raster überlagern.

Bild 6: Das belichtete Ergebnis



Die Gradationskurve verläuft in einem Koordinatensystem. Auf der x-Achse ist von links nach rechts die Helligkeit der "Bildwerte" aufgetragen (von 0 bis 255). Auf der x-Achse liegen die "Zielwerte", ebenso von 0 bis 255. Man kann also die Zuordnung der Zielwerte zu den Bildwerten ablesen. Eine gerade Linie von links unten nach rechs oben bedeutet, daß jeder Zielwert genau so hell ist wie der Bildwert. Ihre Druckvorlage darf also zum Beispiel nur einen Tonwertumfang von 9% zu 88% besitzen. Ihr eingescanntes Bild ist aber durchaus heller als als 9% bzw. dunkler als 88%. Dann können Sie die Gradationskurve so verändern, daß optimale Druckergebnisse erzielt werden können. Der Rechner verteilt die Helligkeitszuordnung um. Er ändert den Kontrast des Bildes (Bild 4).

Als letztes möchte ich noch auf einen Effekt eingehen, der besonders in der EBV gefürchtet ist. Ihnen ist bestimmt schon einmal ein Bild aufgefallen, welches außer dem eigentlichen, abgebildeten Gegenstand lauter kleine helle und dunkle Flächen enthält. Diese Erscheinung nennt man Moiré. Ein Moiré kann entstehen, wenn sich zwei Raster überlagern (Bild 5).

Dies geschieht sehr oft beim Scannen von gedruckten Vorlagen, welche ja ihrerseits bereits gerastert sind. Nachdem der Scanner das Bild seinerseits in ein feines Raster zerlegt, können sehr leicht Interferenzen auftreten. Man kann versuchen, die Vorlage so zu drehen, daß der Effekt möglichst klein bleibt. Dies ist manchmal nicht möglich, da das Bild dann freilich auch gedreht auf dem Bildschirm erscheint und einige Programme nicht in der Lage sind, es wieder "zurückzudrehen". Eine zweite Möglichkeit besteht darin, eine oder zwei Kopierfolien zwischen Vorlage und Scanner zu legen. Das Bild ist dann nicht mehr im optimalen Focus des Scanners, d.h. es wird unschärfer, und das Druckraster erscheint der Optik des Scanners nur mehr als einheitliche graue Fläche.

Nachdem Sie nun all dies beachtet haben, dürfte dem abschließenden Export des gerasterten Bildes (meist im .IMG- oder .CRG-Format) nichts mehr im Wege stehen. Denn schließlich wollen Sie es auf einem Satzbelichter so richtig zur Geltung kommen lassen. Dazu importieren Sie es in Ihren CALAMUS (nachdem es auf dem ATARI keine Alternative zu CALAMUS gibt), optimieren seine Größe für die jeweilige Belichterauflösung, setzen den Begleittext drumherum, speichern das Dokument ab - und haben hoffentlich eine Wechselplatte. Denn der Speicherbedarf eines Bildes in Linotronic-Auflösung sprengt sehr schnell die Kapazität einer Diskette.

Spätestens jetzt wird Ihnen klar, daß ein gutes EBV-System, selbst wenn ein ATARI das Herzstück ist, schnell die 10 Kilo-Mark-Grenze überschreiten kann. Aber wozu hat man denn noch eine Oma...

Das Wechselmedium bringen Sie ins nächste Belichtungsstudio, und kurz danach sind Sie der stolze Besitzer einer High-End-Reproduktion Ihres Rechners. (Bild 6).

Natürlich haben Sie gleich eine zweite Lithografie erstellen lassen, für die andere Oma, versteht sich ...

LR

Bunt is beautiful

Bildbearbeitung in Farbe mit TmS Cranach

Farbretusche und Druckvorlagenherstellung für den Vierfarbdruck - davon wagten wir ATARIaner bis jetzt nicht zu träumen. Neidische Blicke in Richtung Äppelmäc und banges Warten auf neue EBV-Software resultierten aus diesem Umstand. Erste Gehversuche in Sachen Farbbildverarbeitung am ATARI ST erlaubt nun das Programm "Cranach" von der Firma TmS aus Regensburg. Eine Profiversion "Cranach Studio" ist für die ATARI-Hausmesse angekündigt.

TmS Cranach ist nicht sehr anspruchsvoll, was die Hardware anbelangt. Schon spärliche 1 MB Hauptspeicher und ein SM 124 Monitor reichen aus, um mit Cranach arbeiten zu können. Sinnvoll erscheint mir eine solche Konfiguration allerdings nicht: Da Farbbilder immerhin dreimal mehr Speicher benötigen als monochrome Halbtonbilder (je ein Byte für den Rot-, Grün- und Blauanteil eines Bildpunktes), wird der Speicher auch bei kleineren Bildern schnell knapp. Leider erlaubt TmS Cranach keine virtuelle Speicherverwaltung - vier MB Hauptspeicher können demnach bestimmt nicht schaden. So richtig bunt wird es erst, wenn Sie Ihrem ST eine Farbgrafikkarte (z.B. eine MGE-Karte) und einen Farb-MultiSync-Monitor verpassen. Cranach läuft dann komplett auf dem Farbbildschirm. Leider kann bis jetzt keine Grafikkarte alle 16.7 Mio Farben, die Cranach intern verwaltet, darstellen. 256 verschiedene Farben aus einer Palette von 16,7 Mio Farben sind das höchste der Gefühle. Aber dem tristes Grau gewohnten ST-User flimmert auch so eine neue - farbige - Welt vor Augen.

Die Benutzeroberfläche von Cranach erscheint wie die eines jeden "ordentlichen" GEM-Programmes: Alle Werkzeuge und Operationen sind über die Menüleiste zu erreichen. Auch Anfänger finden sich deshalb schnell mit Cranach zurecht. Die wichtigsten Operationen können Sie außerdem durch Tastendruck aktivieren.



Originalbild: Dieses Bild wurde herkömmlich...


Cyan-Auszug


Magenta-Auszug


Yellow-Auszug


Schwarz-Auszug

Ob mit AMIGA, MAC oder PC: Durch die zahlreichen Formate wie TIFF, IFF und GEM-Image versteht sich Cranach blendend mit anderen Systemen. Sogar reine Rasterbilder, denen die Farb- bzw. Grauwertinformation fehlt, können in Bilder mit 24 Bit Tiefe umgewandelt werden. Damit steht den vielen Besitzern eines Raster-Scanners endlich die Welt der Halbtonbilder offen. Freilich erreicht ein solcherart umgewandeltes Bild nicht die Qualität eines Bildes, welches mit einem Halbton-Scanner erzeugt wurde.

Damit sind wir auch schon bei den Bildeingabegeräten: Zusätzlich zu Cranach können Sie den entsprechenden Treiber erwerben, der Cranach die Verständigung mit Ihrem Scanner erlaubt. Lieferbar sind Treiber für alle bekannten Scanner. Nachdem Sie in einem Prescan-Fenster einen Ausschnitt gewählt haben, wird Ihr Bild digitalisiert. Je nach eingestellter Scanner-Auflösung kann der Lesevorgang einige Zeit dauern. Das Bild erscheint dann in einem von sechs möglichen Fenstern, deren Inhalt jeweils über Rollbalken verschoben werden kann. Der Bildaufbau vollzieht sich auch bei Unterstützung durch eine Grafikkarte verhältnismäßig träge. Zudem wiederholt sich die Warterei jedesmal, wenn nach dem Schließen einer Auswahlbox das Bild wiederhergestellt werden muß. Wer schon einmal gesehen hat, wie schnell Grafikkarten Bilder über den Monitor huschen lassen können, wird sich fragen, weshalb diese Option bei Cranach unberücksichtigt blieb. Sei's drum - damit kann man leben - schließlich handelt es sich bei Cranach um das erste grafische Farbsystem für den ATARI ST. Und Trendsetter haben halt immer so ihre Macken.

Zur Manipulierung des Bildes stellt Cranach vielfältige Instrumente zur Verfügung. Alte Bekannte wie Stempel, Wasser, Pinsel, Finger, Spray, Lack und Kreide finden sich ebenso wie Funktionen zum Pausen, Aufhellen und Abdunkeln. Allen Werkzeugen gemein ist, daß der Wirkbereich eingestellt werden kann. So kann ein Farbintervall definiert werden, auf dem das betreffende Werkzeug wirken soll. Außerdem ist es möglich, bestimmte Bildbereiche zu maskieren, d.h. sie werden komplett vor etwaigen Veränderun-

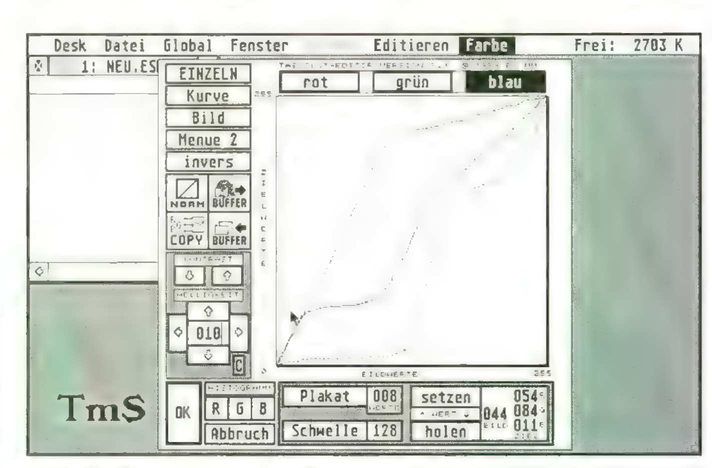
...und dieses mit Cranach separiert

gen bewahrt. Die Malfarbe läßt sich entweder über eine Auswahlbox manuell - durch Eingabe der drei RGB-Werte - oder durch Aufnehmen aus dem Bild definieren. Die Auftragstärke ist bei allen Werkzeugen frei wählbar - von 1% bis 100%, transparent oder deckend, wie's beliebt. Haben Sie dem aktuellen Bild einen Undo-Puffer spendiert, können Sie das Bild mit der Funktion "pausen" an einer beliebigen Stelle lokal restaurieren.

Die umfangreichsten Möglichkeiten von Cranach verbergen sich hinter dem Menüpunkt *Collage*. Hier können Sie Bilddteile nach Herzenslust drehen, kopieren, verschieben und skalieren. Den Bildbereich selektieren Sie durch einen einfachen Rahmen, einen Kreis oder durch einen beliebig geformten Polygonzug. Dabei kann auch hier der Deckungsgrad des Objektes eingegeben werden. Außerdem besteht die Möglichkeit, den Bildbereich farblich an die neue Umgebung anzupassen. Aus welchen Gründen auch immer die Entwickler auf eine Zoom-Funktion verzichtet haben - das detailgenaue Ausschneiden von kleineren Bildbereichen ist nahezu unmöglich. Wirklich schade - denn die Collagefunktionen von Cranach eröffnen ganz neue Perspektiven in der Farbbildmontage.

Für die Erstellung von Druckvorlagen ist es meist nötig, die Gradation bzw. die Farbwertverteilung des Bildes zu ändern. Bei Cranach geschieht dies mit Hilfe der CLUT (Color-Look-Up-Table). Dabei gibt es nicht nur eine Gradationskurve wie bei monochromen EBV-Programmen, sondern derer drei. Die Farbwertverteilung eines jeden Farbkanals kann hier komfortabel geändert werden. Bei Verwendung einer Farbgrafikkarte lassen sich die Veränderungen sofort am Bild verfolgen, da es nicht ganz vom Clut-Editor-Fenster überdeckt wird.

Die für die Bildschirmdarstellung verwendete RGB-Trennung kann für den Druck nicht verwendet werden. Hier verwendet man ein Modell mit vier Farbauszügen: Cyan, Magenta, Gelb (Yellow) und Schwarz - das CYMK-Farbsystem. Cranach bietet die Möglichkeit an, ein Bild vom RGB-Modell in das CMYK-Modell umzurechnen. Dabei finden wichtige Parameter Berücksichtigung. So können Sie den maximalen Farbauftrag und die Tiefenkorrektur festlegen sowie bestimmen, ab welchem Grauwert, der durch die Mischung aus den drei Farben



Die CLUT-Einstellung bei TmS Cranach

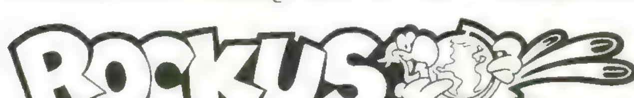
(CMY) entsteht, ein Schwarzanteil(K) hinzugefügt werden soll. Das alles ist nicht ganz einfach zu verstehen - schon gar nicht von einem blutigen "Farbanfänger". Die separierten Farbauszüge können als Halbtonbilder auf Festplatte abgelegt werden. Danach ist jeder Farbauszug einzeln nachbearbeitbar. So kann man sich Schritt für Schritt an ein optimales Druckergebnis herantasten. Um sich von der Qualität des Bildes zu überzeugen, ist es angebracht, das Ergebnis mit Hilfe eines Proof-Ausdruckes zu kontrollieren. Erst dann sollte man die endgültigen Farbauszüge belichten lassen. Da Cranach das Rastern komplett anderen Programmen überläßt, hängt das Ergebnis stark von den verwendeten Raster-Algorithmen ab. Die Bilder in diesem Bericht wurden mit Cranach separiert und von einem Calamus mit Rastermodul aufgerastert und belichtet.

Wie gesagt: Optimale Ergebnisse lassen sich bestimmt nicht aus dem Stand erreichen. Bis zur originalgetreuen Reproduktion eines Farbbildes ist es ein langer und steiniger Weg. Zu viele Parameter und Faktoren sind zu beachten, um sofort zufriedenstellende Ergebnisse zu erzielen. Wenn Sie sich in das Reich der Farbbildverarbeitung wagen wollen, werden Sie kaum an intensiver Lektüre diverser Fachliteratur vorbeikommen. Das Werkzeug ist inzwischen vorhanden - trotz einiger gravierender Mängel bietet Ihnen Cranach die Möglichkeit, Bilder zu separieren und zu bearbeiten. Sicherlich wird die große "Studio"-Version diese Nachteile nicht mehr aufweisen. Die neue Oberfläche ist der von Calamus nachempfunden, auch detailgenaues Arbeiten wird mit der Zoom-Funktion möglich sein - Cranach Studio wird eine wichtige Bereicherung auf dem immer professioneller werdenden DTP-Sektor am ATARI. Für diejenigen, die jetzt schon fleißig "üben" wollen, bietet Cranach für etwa 370 Mark auf jeden Fall eine Fülle von Möglichkeiten, um die nötigen Erfahrungen zu sammeln.

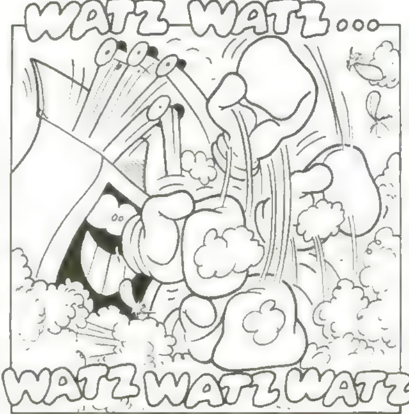
LR

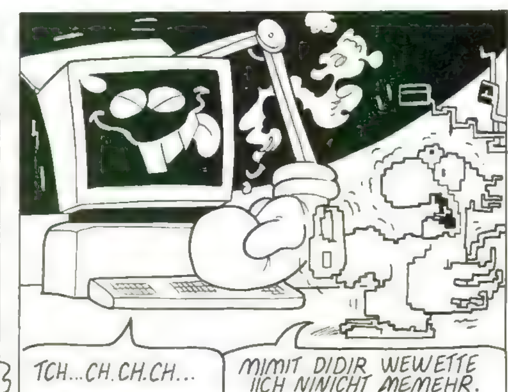
Bezugsadresse:

TmS GmbH
Cranachweg 4
8400 Regensburg
Tel.: 0941/95163











PKS **EDIT**, der Texteditor für gehobene Ansprüche. Trotz spielend einfacher Bedienbarkeit ein mächtiges Werkzeug, welches besonders für Programmierer neue Perspektiven in der Bearbeitung von Texten eröffnet.

- GEM orientierter Editor mit superschnellem Bildschirmaufbau.
- max. 7 Dateien gleichzeitig bearbeitbar.
- umfangreiche Such- und Ersetzfunktionen. Endlich in einem GEM-Editor: Reguläre Ausdrücke für die Definition beliebiger Suchmuster. Strings und Zeichenklassen etc. können als Such- und Ersetzmuster beliebig gruppiert und wiederholt werden.
- Komplette Disketten oder Pfade können durchsucht werden (...wie hieß die Datei mit dem Brief an Herrn Müller ?).
- Komfortable Textblockmanipulationen, Kopieren, Löschen, Verschieben von Blöcken – auch in Spaltenform.
- UNDO für alle Funktionen !!!
- Freies Zeilenformat, dadurch Editieren von allen Dateiformaten und sogar Binärdateien möglich.
- Einfache Erstellung von Makros, Abkürzungen, und Tastenprogrammen. Eingebauter Ereignisrecorder ! Aufruf aller Funktionen sowohl über die Tastatur als auch mit der Maus. Tasten frei belegbar.
- PKS **EDIT** arbeitet mit jedem Compiler zusammen und springt in fehlerhaften Programmtexten automatisch in die entsprechenden Zeilen.
- Erstellen und Verwalten von Kreuzverweislisten, Überprüfung von Klammerschachtelungen.
- Mehrere Bearbeitungsmodi: Autoindent, Autowrap, Tabulatorpositionen beliebig, expandieren und komprimieren von Tabulatoren, u.v.m...
- Übersichtlicher Vergleich von Dateien.
- Mächtige Sortierfunktion erlaubt auch das Sortieren von Datenbanksätzen...
- Eingebaute Shell mit Löschen und Umbenennen von Dateien. Aufruf von Programmen mit Parametern, erweiterte Bourne-Shell mit MAKE und vielen Dienstprogrammen verfügbar.
- Ausdruck von Dateien, Druckertreiber kompatibel mit WORDPLUS.

...aber das ist noch nicht alles, fordern Sie gleich die Demodiskette an (10.- DM als Scheck etc.), um PKS **EDIT** kennenzulernen.

empf. Preis mit Handbuch DM 148.-



Pahlen & Krauß Software
Dieffenbachstr. 32
1000 Berlin 61
Tel. 030 – 786 59 45

DER ETWAS ANDERE VERSAND!

24-Stunden Service!

Wir garantieren, daß jede Bestellung spätestens 24 Stunden nach Eingang unser Haus verläßt, sofern verfügbar. Auf alle gekauften Artikel erhalten Sie natürlich volle Garantie. Wir führen jede verfügbare Hard- und Software für den Atari ST, sowie alle Bücher. Hier ein kleiner Auszug aus unserem reichhaltigen Programm:

NEUHEITEN:
Combo Racer 80,-
Damucles 80,-
Imperium 85,-
Last Ninja 2 80,-
Logo 85,-
Magic Lines 60,-
Powerboat 80,-
Qatbol 50,-
Resolution 101 80,-
Rotox 60,-
Tusker 80,-

SPIELESOFTWARE:
Balance of Power 1990 75,-
California Games 55,-
Castle Master 75,-
Chaos Strikes Back 80,-
Dragonflight 85,-
Dungeon Master 75,-
Elite 65,-
Emlyn Hughs Intern. Soccer 80,-
Esprit 95,-
F-16 Falcon 80,-
F-16 Mission Disk I 65,-
F-19 Retaliator 95,-
Flight Simulator II deutsch 85,-
jede Scenery Disc dazu 45,-
FS II Disk Hawaiian Odyssey 45,-
Great Courts 85,-
Indiana Jones, Adventure 75,-
Italy 1990 Winners Edition 60,-
Jumping Jack Son 60
Kaiser 120,-
Kick off II 65,-
Klax 55,-
Leisure Suit Larry 80,-
Leisure Suit Larry II 95,-
Leisure Suit Larry III 115,-
Maniac Mansion 85,-
Manchester United 45,-
Midwinter 85,-
Öl Imperium 65,-
Pirates 80,-
Player Manager 65,-
Populous 85,-
Psion Chess 75,-
Rainbow Islands 55,-
Rings of Medusa 85,-
Rorke's Drift 75,-
Sim City 85,-
Sonic Boom 80,-
Space Quest III 95,-
Tie Break 85,-
Tennis Cup 85,-
Zak McKracken 75,-

ATARI POWER PACK
Compilation mit 20 Super-Spielen z.B. Gauntlet II, Outrun, Space Harrier, Starglider, Afterburner 135,-

UNSER TIP DES MONATS:
Alternative Mäuse:
- Logi-Maus Pilot 100 % ST-kompatibel incl. Utility-Software „Pilot Control" sowie Spiel „Pipe Mania" (3 Level) 95,-
-Reis-Ware-Maus, 100 % ST-kompatibel 85,-
- Marconi Trackball, die Alternative zur Maus, hervorragend geeignet für DTP- und CAD-Arbeit 195,-

Disketten: 10 Stck. Packungen
3,5" NO NAME MF2DD 10,-
3,5" BOEDER 2DD farbig 25,-
5,25" NO NAME MD2D 5,-

CAD/Graphik:
Arabesque 275,-
GFA-Chemgraf 75,-
GFA-Draft plus 340,-
Systembibliotheken dazu je 145,-
IPA Degenis III 165,-
Omikron DRAW 125,-
STAD 175,-

DTP/Textverarbeitung/Editoren
CAD 3D Cyber Studio 175,-
Calamus Outline Art 395,-
Edison 165,-
PKS-Write 195,-
Script 195,-
Signum! 2.0 440,-
unsere Zusatzprogramme für Signum!/STAD
Headline, Großschriften 95,-
Fontmaker 95,-
Convert 95,-
SDO merge 50,-
SDO index 50,-
SDO preview 50,-
SDO graph 50,-
MetaxMap 50,-
Tempus 2.0 125,-
That's Write 345,-
1st Word plus 3.15 245,-
1st Proportional 115,-

Datenbanken/Tabellenkalkul.:
Adimens 3.0 Plus 395,-
LDW-Power Calc 245,-
Steuer Tax '89 95,-
That's Address 185,-
Themadat 245,-
1st Address 75,-

Programmiersprachen:
Omikron Compiler 175,-
ST Pascal plus 245,-
Turbo C 2.0 .... ab 245,-

Utilities:
Anti Virus Kit 95,-
BTX-Manager 3.02 385,-
Copy Star 3.0 165,-
Epsimenu 85,-
Hotwire 75,-
Interlink 75,-
Mortimer 75,-
Multidesk 75,-
Neodesk 2.05 85,-
Revolver 75,-
Turbo ST 1.8 85,-

MS-DOS auf dem Atari ST:
PC-Speed 395
AT-Speed 545,-

Zubehör:
Staubschutzhauben Kunstl. für:
ATARI SM 124 30,-
ATARI 1040 o. Mega Tast. je 20,-
ATARI 260/520 ST 15,-
Mega ST Set Monitor + Tast. 50,-
andere Monitore + Drucker .. a.A.
Mausmatte 15,-
Media Box 3,5" f. 150 Disks . 40,-

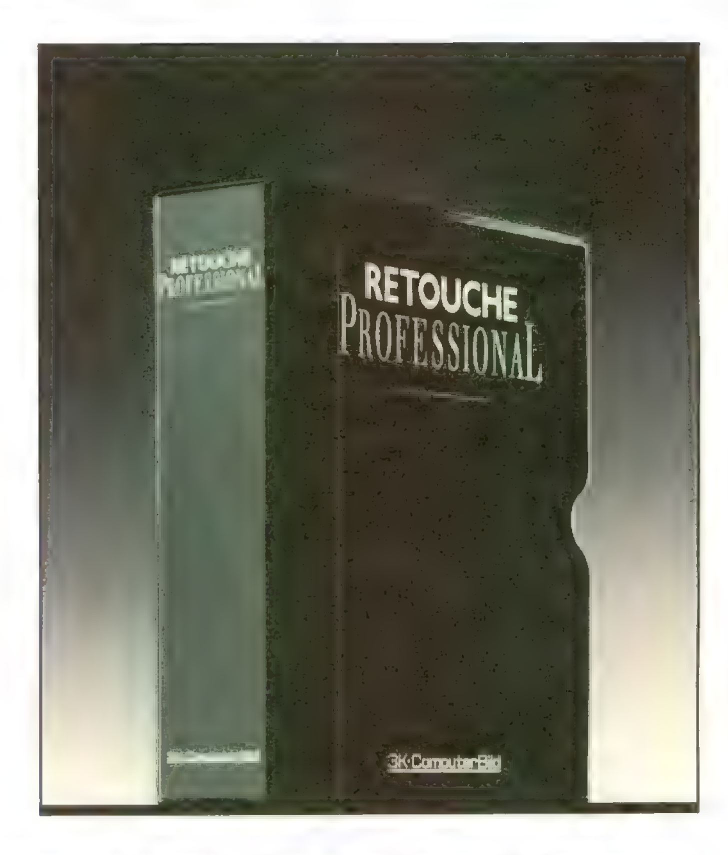
Hardware:
Monitorumsch. o. Reset .. ab 50,-
NEC P2 plus Drucker 795,-
Profile 20 Festplatte 798,-
Profile 30 Festplatte 950,-
Profile R44 Wechselplatte 1998,-

PUBLIC DOMAIN:
Wir führen alle PD- Disketten der verschiedenen Serien für den ATARI ST.
JEDE DISKETTE nur 5,- DM

Kostenlose Kataloge für PD. Bücher, Hardware und Software bitte getrennt unter Angabe Ihres Computertyps anfordern. Lieferung per NN zzgl. 7,- DM Versandkosten. Bei Vorauskasse zzgl. 3,- DM, ab 100,- DM Bestellwert versandkostenfrei. Auslandsversand grundsätzlich zzgl. 15,- DM Versandkosten. Auf Wunsch auch UPS-Versand.

COMPUTER-VERSAND
Schlichting
...der etwas andere Versand

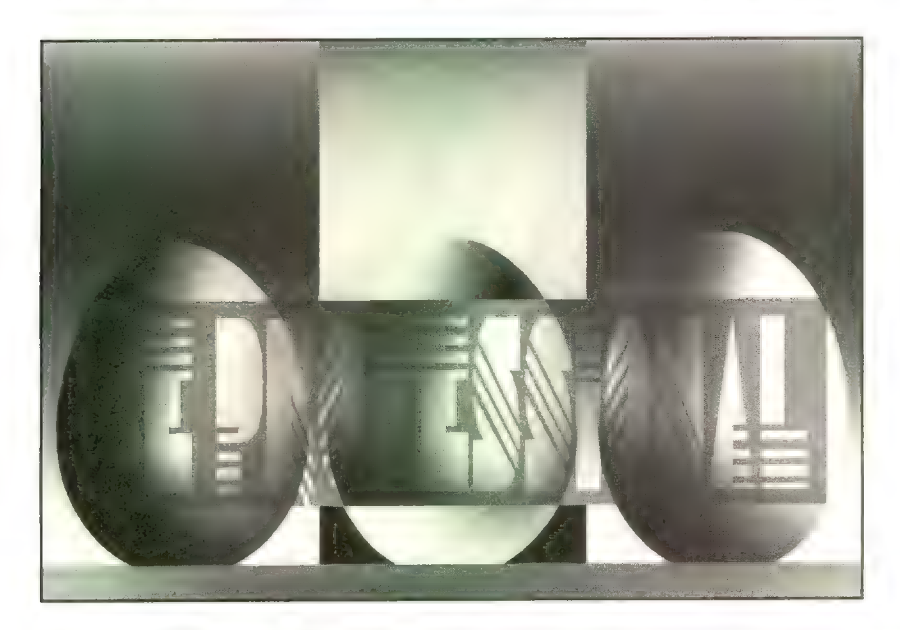
Rund um die Uhr: ☎ 030 / 786 10 96
Playsoft-Studio-Schlichting
Computer-Software-Versand GmbH
Postanschrift / Ladengeschäft: Katzbachstraße 8
D-1000 Berlin 61
Fax: 030/786 19 04 · Händleranfragen erwünscht



Werkzeug für Profis

RETOUCHE PROFESSIONAL

Grafik-Profis am ST hatten bis jetzt nicht das geeignete Werkzeug, um professionelle Schwarzweißreproduktionen herzustellen. Zwar gab und gibt es das Programm RETOUCHE. Doch durch das eingeschränkte Bildformat von 640 x 400 Bildpunkten war an hochwertige Lithografien in Größen bis DIN A2 nicht zu denken. Außerdem rief die (noch) sehr kleine Gemeinde der "ATARI-Lithografen" nach Software, die die Leistungen von Grafikkarten nutzen kann. Ihr Rufen wurde erhört: RETOUCHE ist endlich erwachsen geworden! RETOUCHE PROFESSIONAL macht seinem Namen alle Ehre.



Geliefert wird RP in einem edlen schwarzen DIN A4-Ordner, satte drei kg schwer. Die Software befindet sich auf drei Disketten - einer Programmdiskette und zwei Discs mit Bildern. Die einzelnen Funktionen des Programms werden sehr detailliert erklärt, allerdings wäre ein Tutorial wie bei der kleinen Version von Retouche durchaus nicht fehl am Platze.

Bevor Sie sich die einzelnen Programmpunkte ansehen, werfen wir einen Blick auf den konzeptionellen Aufbau von RP: RP ist ein professionelles Programm. Die Entwickler sind davon ausgegangen, daß der durchschnittliche Anwender über eine Grafikkarte und einen entsprechenden Bildschirm verfügt. Sie können mit RP auch nur mit dem SM 124 arbeiten - allerdings sehen Sie das Bild dann nur gedithert. Professionelle Retusche von Halbtonbildern setzt voraus, daß Sie diese auch als solche sehen können. Deswegen arbeitet Retouche Professional mit gängigen Grafikkarten (z.B. MGE) zusammen. Sie sehen die gescannte Vorlage also mit 256 Graustufen. Das Bild wird zum Teil oder ganz, je nach Größe, im internen Speicher der Grafikkarte abgelegt. Dadurch kann es nahezu blitzartig hin- und hergeschoben werden. Auch das Zoomen geht atemberaubend schnell. Außerdem hat die Arbeit an einem Graustufenmonitor einen weiteren gewichtigen Vorteil: Alle Bedienungselemente, Parameter und Werkzeuge werden auf dem SM 124 dargestellt, auf dem Graustufenschirm befindet sich das Bild in seiner vollen Größe. Ein Druck auf die Escape-Taste schaltet zwischen den beiden Schirmen um: Der Mauszeiger springt einfach von einem in den anderen; arbeiten Sie nur mit einem Monitor, muß zuerst die "Werkzeugseite" gelöscht und dann die Grafikseite aufgebaut werden. Das "Zwei-Bildschirm-Konzept" erlaubt also extrem zügiges Arbeiten.

Man merkt, daß die Entwickler dem Punkt Bedienerführung sehr viel Aufmerksamkeit gezollt haben. Zwar erscheint der Werkzeugbildschirm auf den ersten Blick enttäuschend leer: Die Miniaturen der maximal 10 Bilder, die RP verwalten kann, befinden sich in der linken oberen Ecke. Rechts davon finden Sie das Auswahlmenü für die verschiedenen Stiftgrößen und -formen. Die rechte obere Ecke wird mit dem Übersichtsfenster ausgekleidet. Im unteren Bereich der Seite



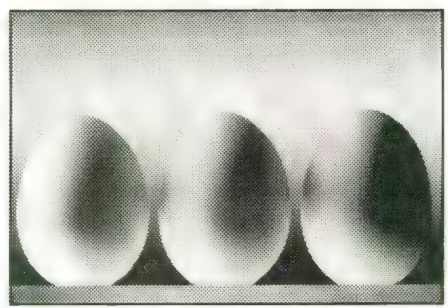
So präsentiert sich Retouche Professional auf dem Werkzeugschirm

sind Graukeil und die verschiedenen Werkzeuge angesiedelt. Wie gesagt, auf den ersten Blick etwas wenig, aber beim Anklicken der einzelnen Funktionen erscheint direkt neben dem Mauszeiger ein Pop-Up-Menü, welches je nach Funktion bis zu zehn Untermenüs besitzen kann. Haben Sie eine Funktion bereits einmal angewählt, steht der Mauszeiger bei erneutem Aufruf bereits auf dem entsprechenden Eintrag. Viele Funktionen können auch direkt über die Tastatur eingegeben werden. Dadurch entfällt sogar das Umschalten auf den Werkzeugbildschirm. Kurz gesagt: Das Arbeiten mit RP wird auch eingefleischte GEM-Menü-Fans oder Pictogramm-Fanatiker überzeugen.

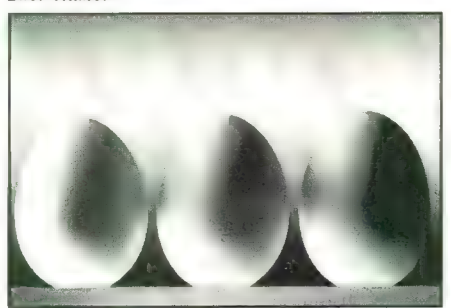
Wie gesagt, RP kann bis zu 10 verschiedene Bilder gleichzeitig verwalten. Diese können jeweils bis zu 4096 x 4096 Bildpunkte groß sein. Sie sehen schon, daß mit herkömmlichen Methoden der Bildorganisation nicht gearbeitet werden kann, würde doch schon ein solches Bild knapp 16 MB beanspruchen. Also haben die Entwickler von RP die virtuelle Speicherverwaltung implementiert. Ein virtueller Speicher ist ein nur scheinbarer, ein gedachter Speicher zur Ergänzung des Arbeitsspeichers. Er befindet sich auf der Festplatte und überträgt seinen Inhalt in den Arbeitsspeicher, wenn er dort benötigt wird. Die Geschwindigkeit der Bildbearbeitung hängt dann in hohem Maße von der Geschwindigkeit und Kapazität der Festplatte ab.

Für jedes Bild können Sie einen Undo-Puffer einrichten, praktisch eine Kopie des Bildes. Auf Tastendruck wird dieser aktualisiert bzw. mit dem Bild vertauscht, um "Patzer" auszubügeln. Freilich benötigt auch der Undo-Puffer genauso viel Speicher- bzw. Festplattenplatz wie das Bild selber.

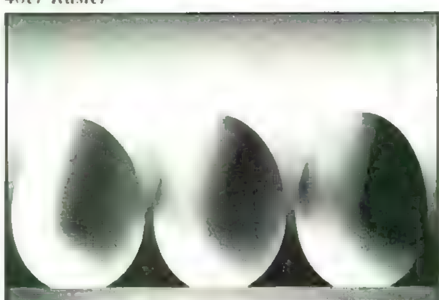
Auch beim Import und Export von Bildern ist RP sehr flexibel. In erster Linie wird ein Scanner das Programm mit Bildmaterial versorgen. Außerdem können alle gängigen ST-Farbgrafikformate geladen und grauwertrichtig umgesetzt werden. "Leib- und Magenformat" von RP ist das TIFF-Format, Standard auf PC und MAC. Exportieren können Sie mit Hilfe des Rasterexport-Moduls. Es rastert Ihre Bilder entprechend der gewählten Auflösung (z.B. 300 dpi für Laserdrucker oder



28er Raster



48er Raster



62er Raster

Mit Retouche Professional können Sie beliebig grobe Raster erzeugen. Vom 20er für den Laserdrucker bis zum 120er für Trockenoffsetdruck.

bis zu 2540 dpi für einen Linotronic-Belichter) auf und speichert sie im .IMG- oder .CRG-Format. Wundern Sie sich nicht über die Größe der Datei - bei 2540 dpi kann die Festplatte schnell voll sein!

Die zweite Export-Möglichkeit ist die Ausgabe als PostScript-Datei oder die Ansteuerung von PostScript-Geräten. Außerdem übernimmt das Exportmodul von RP die Ansteuerung des ATARI Lasers SLM 804.

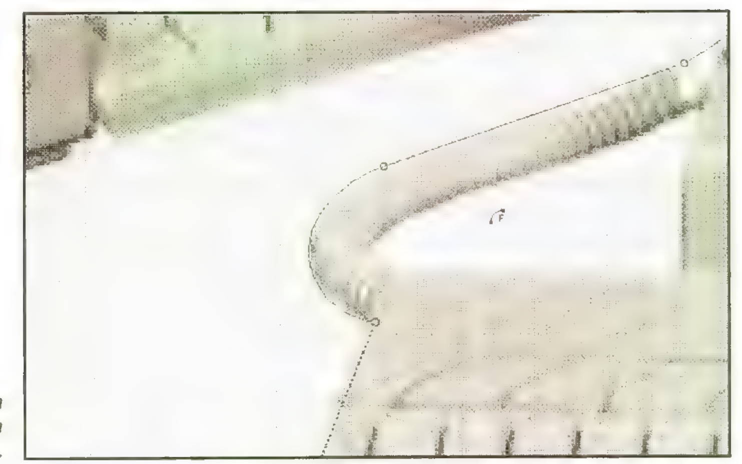
Das Rastermodul greift auf eine Rasterbibliothek mit über 100 verschiedenen Rasterweiten und -winkeln zurück. Diese Raster wurden übrigens mit viel Mühe von Hand optimiert und gewährleisten einen wirklich linearen Grauwertanstieg. Mit Hilfe des Rasterexportmoduls ist auch die sog. Endrasterkontrolle möglich, d.h. man kann selbst bei einem 100er-Raster bei 1000 L/cm jeden einzelnen Rasterpunkt betrachten.

Eine weitere Novität bei Bildverarbeitungsprogrammen ist die Verwendung von Vektorpfaden.

Diese bieten sich immer dort an, wo auf einem bestimmten Weg (z.B. einem Umriß) ein Werkzeug mehrfach oder verschiedene Werkzeuge nacheinander eingesetzt werden. Von Hand ist es nahezu unmöglich, den gleichen Weg mehrfach exakt abzufahren - auf einem einmal angelegten Pfad dagegen ist dies kein Problem. Das Anlegen der Pfade ist sehr einfach möglich. Durch ein paar Mausklicks legt man Linien und Bézierkurven an. Diese haben keine - wie z.B. bei Outline Art oder MegaPaint - Stützpunkte. Die Linie geht direkt durch die vier gesetzten Punkte. Um den Verlauf der Kurve zu ändern, "fassen" Sie einfach mit dem Mauszeiger irgendwo in der Kurve an und trimmen Sie zurecht. So habe ich mir immer den Umgang mit Bézierkurven gewünscht.

Um es vorwegzunehmen: Da RP über eine derart reichhaltige Auswahl an Werkzeugen verfügt, ist es an dieser Stelle nicht möglich, detailliert auf jedes einzelne einzugehen. Deswegen möchte ich Ihnen nur einen groben Überblick verschaffen.

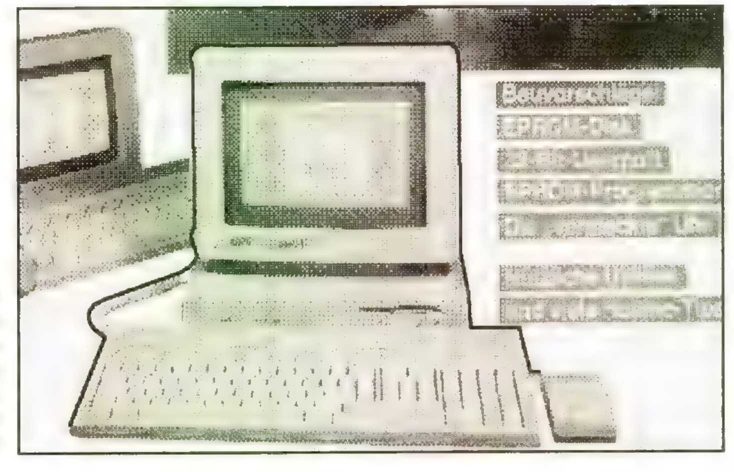
Die meisten Instrumente wie Stift, Kreide, Stempel, Wasser und Finger sind Aufsteigern vom "kleinen" Retouche schon bekannt. Ebenso wie bei der Urversion können Sie bei RP den Andruck, d.h. die Intensität der Wirkung, meist zwischen 1 und 10 verstellen. Neu bei RP sind der Schärfer und der Kopierstift. Mit dem Schärfer werden Kontraste erhöht, er ist das Gegenstück zur Funktion *Wasser*. Auch der Kopierstift ist eine feine Sache: Zuerst definieren Sie einen Quellpunkt in einem beliebigen Bild. Der erste Punkt, den Sie im Zielbild anklicken, wird zum Referenzpunkt, und relativ zu diesem Referenz- wird vom Quellpunkt her kopiert. Eine Arbeitserleichterung sind auch die Intrumente Densitometer zur Helligkeitsbestimmung und das Maßband.



Auf einem einmal angelegten Vektorpfad...

Der Umriß des Bézierblocks kann automatisch in einen Vektorpfad umgerechnet werden, so daß Sie nach dem Einkopieren in ein anderes Bild leicht z.B. mit der Funktion Wasser die Konturen verwischen können.

RP verfügt auch über eine mächtige Funktion zur Erzeugung beliebiger Grauverläufe. Zum einen kann ein Block oder ein Bézierblock mit verschiedenartigen Verläufen gefüllt, zum anderen können zwei Vektorpfade mit beliebigen Verläufen verbunden werden. So ist es möglich, einen Grauverlauf auch komplizierten Umrissen anzupassen.



...kann man verschiedene Werkzeuge entlangfahren lassen. Hier z.B. den Maskierstift



...projiziert,..

Ein wichtiges Hilfsmittel bei elektronischen Bildverarbeitungssystemen ist die sog. Maske. Mit ihr werden Bildbereiche, die vor einer Bildmanipulatuion "bewahrt" werden sollen, einfach abgedeckt. Bei RP ist die Maske, sofern Sie mit einer Grafikkarte arbeiten, farbig und transparent. So sieht man zwar noch den verdeckten Bildbereich, trotzdem ist er vor "unerlaubtem Zugriff" geschützt. Selbstverständlich kann die Maske automatisch auf bestimmte Helligkeitsbereiche gesetzt werden - eine großartige Hilfe z.B. beim Freistellen von Bildteilen. Wie jedes Werkzeug kann auch der Maskierstift entlang eines Vektorpfades laufen. So kann man sehr schnell maskieren, ausschneiden, einpassen und Kanten verwaschen - die Werkzeuge laufen wie von Geisterhand entlang des Pfades.

Mit den mächtigen Blockfunktionen können Sie nun endlich Blöcke beliebig in Größe und Form variieren. Sie können Blöcke verzerren, drehen, projizieren und dreidimensional verformen. Der ursprünglich rechteckige Bildbereich, den Sie als Block definieren, kann auf sog. Bézierblöcke projiziert werden. Ein Bézierblock ist ein Liniennetz, welches an den Kreuzungspunkten zweier Linien verzerrt werden kann. Auf ein derart gestaltetes Netz wird dann der Quellblock projiziert.



Quellblock...



...und mit der Bézierblockfunktion...



...verzerrt,..



...verformt.

Die Effekte wie Schärfen, Konturieren, Aufweichen, Aufrauhen, Strukturieren etc. erklären sich selbst. Sie kennen Sie zum Teil bereits aus Retouche.

Gradationsveränderungen lassen sich, wenn Sie mit einer Grafikkarte arbeiten, sofort auf Ihrem Graustufenschirm beobachten. Die Kurve selbst können Sie einer edierbaren Bézierkurve anpassen. Mit Hilfe der Funktion *Histogramm errechnen* können Sie den Grauwertumfang Ihres Bildes umverteilen: gewichtete Gradation, zentrierte Gradation - je nach Bedarf korrigieren Sie Kontrast und Helligkeit des Bildes.

Entspricht Ihr Bild nun nach all den Manipulationen Ihren Vorstellungen, lassen Sie es mit Hilfe des Ausgabemoduls aufrastern, um es später in Calamus importieren zu können, von wo es dann zu Film gebracht werden kann. Da Calamus bis jetzt noch keine virtuelle Speicherverwaltung beherrscht, müssen Sie sich auf eine Bildgröße von knapp 3 MB beschränken. Deshalb bietet die Firma 3K ab September eine Satzbelichterversion von RP an. Sie wird zusammen mit einer Grafikkarte mit 1MB Speicher angeboten, die bei der Rasterung sehr viel Zeit einsparen hilft. Der schnelle Grafikprozessor greift dem M68000 des ST gehörig unter die Arme. Die Bilderzeugung von bis zu 2 Mio. Pixeln pro Sekunde ist somit erheblich schneller als übliche PostScript-Rips. Da diese Version von RP den Satzbelichter direkt ansteuern kann, lassen sich Rasterbilder von bis zu 16MB direkt ausbelichten.

Für diejenigen, die es ganz eilig haben, wird erstmals auf der ATARI-Messe ein Hell-Satzbelichter mit 3K-Computerbild-Online-RIP vorgestellt. Das System erlaubt dann erstmals Rasterausgabe von Halbtonbildern ohne Zeitverzug durch Online-Rasterverfahren. Das ganze System inkl. Belichter wird dennoch unter 50.000 DM kosten. Man darf also gespannt sein!

Ab Ende September werden ein Accessory zum Einbinden von Calamus-Fonts in RP und eines für spezielle Bildeffekte für jeweils 398 Mark angeboten.

Fazit: RP ist ein überaus durchdachtes Programm. Die mächtigen Funktionen gehen weit über die Möglichkeiten der herkömmlichen Lithografie heraus. Die Benutzeroberfläche ist optimal an die speziellen Anforderungen bei der EBV angepaßt. Durch den guten Kundensupport im Hause 3K-Computerbild ist eine Beratung auch nach dem Kauf gewährleistet. Für einen Preis von 1198 Mark erhält der Käufer ein professionelles Werkzeug, das auch die schwierigsten Aufgabenstellungen problemlos bewältigt.

LR

Bezugsadresse:

3K-Computer-Bild GmbH
Sassenfeld 71
4054 Nettetal 1
Tel.: 02153/60001

Publishing Partner Master



Professionelles Desktop Publishing auf dem ST ?

Publishing Partner? Moment, das Programm kennen wir doch schon! Ja, richtig, schon im April 1987 haben wir es testen können, damals in der Version 1.01. Doch seitdem sind einige Jahre ins Land gegangen. Haben Sie für das Programm etwas Neues gebracht? Diese Frage wollen wir im folgenden beantworten.

Geschichte

Bereits Ende 1986 konnte man die Urversion von Publishing Partner erstehen. Schon damals hatte der Programmierer (jawohl, nur einer) Erstaunliches geleistet. Immerhin unterstützte bereits diese Version die Seitenbeschreibungssprache PostScript, mit der auch verschiedene Satzmaschinen arbeiten. Als Hauptmanko hoben wir damals die fehlende Trennhilfe hervor, die oft zu unmöglich großen Zwischenräumen führte, und die fehlende UNDO-Funktion. Ein weiteres großes Manko waren die häufigen Abstürze. Publishing Partner konnte sich jedenfalls in Deutschland nicht durchsetzen, zumal es in Calamus einen wirklich deutschsprachigen Konkurrenten hatte. Es wurde ziemlich ruhig um das Produkt.

In den USA ging die Entwicklung jedoch weiter. Mitte 1988 erschien eine neue Version des Programms, vollständig überarbeitet und unter dem neuen Namen PageStream, da der ursprünglich gewünschte Name Publishing Partner Professional aufgrund eines Rechtsstreites mit einer weiteren Software-Firma nicht gewählt werden durfte. Dieses neue Programm machte eine stürmische Entwicklung durch. Ende 1988 war es bereits in der Version 1.51 zu haben. Auf diese Zwischenversion konnte ich auch einen Blick werfen. Seit 1989 ist PageStream (= Publishing Partner Master) auch für den AMIGA erhältlich.

Nachdem für die Version 1.5x kein Distributor für Deutschland gefunden werden konnte - verschiedene Firmen hatten abgelehnt, weil das Programm in ihren Augen immer noch zu absturzgefährdet war -, wurde Mitte 1989 eine Kooperation mit Upgrade Systems in Paris vereinbart. Diese Firma stellte das Programm dann auch 1989 auf der ATARI-Messe in Düsseldorf vor, blieb allerdings ziemlich unbeachtet.

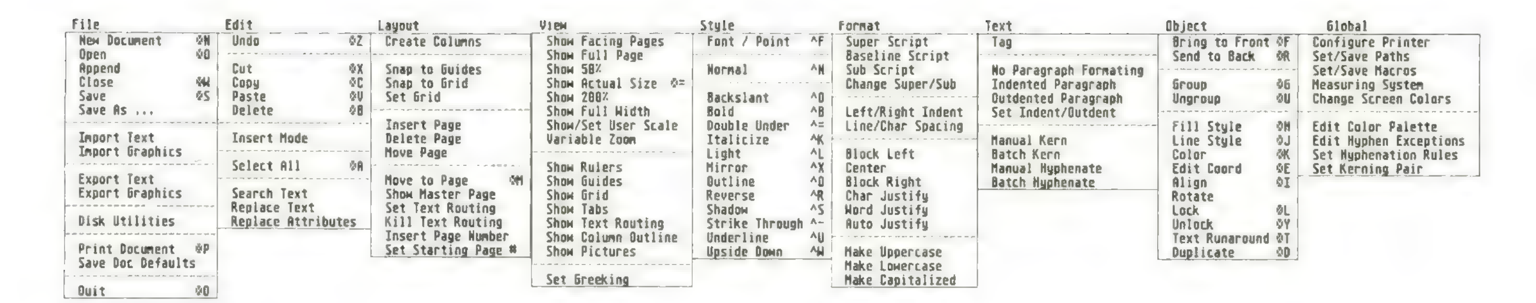
Inzwischen ist man mit der Versionsnummer bei 1.80 angelangt, wobei man allerdings zwischen einer Euro- und der US-Version unterscheiden muß. In der Euro-Version fehlt zur Zeit noch die Rechtschreibhilfe. Ein deutscher Vertrieb hat sich inzwischen auch gefunden, so daß in Kürze mit einem vollständig eingedeutschten Programm zu rechnen ist.

Testpark

Zum Test zur Verfügung stand die englischsprachige Version von Publishing Partner Master (1.80). Zum Lieferumfang gehören 4 doppelseitige Disketten, von denen der Löwenanteil mit Zeichensätzen belegt ist (mehr als 2 MByte), ein Handbuch im Schuber und eine Registrierkarte. Außerdem lag noch ein rosafarbenes Addendum zum Handbuch bei, das die Installation des Programmes erklärte. Das Programm ist nicht kopiergeschützt. Getestet wurde auf einem MEGA 4 mit 80 MB-Festplatte, und die Ausdrucke wurden auf einem Epson LQ-850 und einem Fujitsu PostScript-Laserdrucker gefertigt.

Was man schwarz auf weiß besitzt

Das Handbuch von Publishing Partner Master (PPM) hat einen aureichenden Umfang, auch wenn es stellenweise sehr knapp wirkt. Es ist in 13 Kapitel untergliedert, wobei in der Einleitung erklärt wird, was ein Anwender lesen sollte, je nach



Erfahrungsgrad in DTP und im Umgang mit Vorläuferprodukten von PPM. Leider fehlt ein Stichwortverzeichnis; es gibt zwar einen Index nach Menüfunktionen, der immerhin schon eine gewisse Hilfe ist, aber ansonsten ist man auf sein Gedächtnis angewiesen, wenn man spezifisch etwas nachschlagen möchte. Das Handbuch muß man auf jeden Fall einmal durchblättern, da an vielen Stellen wichtige Tips zum Umgang mit der Software gegeben werden. Neben einer sehr guten, allgemeinen Einführung in DTP und wichtigen Hinweisen am Ende enthält das Handbuch auch zwei Übungsabschnitte, die einem einen Überblick über die vorhandenen Funktionen und die Möglichkeiten des Programms vermitteln. Ansonsten ist das Handbuch ein Beispiel für übersichtliches und sauberes Layout und gut strukturierte Darstellung von Zusammenhängen.

Wann geht's los?

Bevor man Publishing Partner Master das erste Mal starten kann, muß es installiert werden. Das Programm benötigt mindestens 1 Megabyte RAM und ein doppelseitiges Laufwerk. Man kann wirklich mit 1 MByte Speicher vernünftig arbeiten, aber insbesondere beim Drucken sind erheblich längere Ausgabezeiten in Kauf zu nehmen. Das Programm läuft auch auf Großbildschirmen, was sofort überprüft wurde. Es gab keine Probleme bei der Verwendung von Overscan, der größere Bildschirm wurde auch sofort richtig ausgenutzt. Obwohl PPM auch in der mittleren Auflösung arbeitet, wird die Verwendung eines monochromen Monitors empfohlen.

Anwender, die eine Festplatte ihr eigen nennen, können die Installation automatisch durchführen lassen, es werden dann rund 2,7 MByte Platz auf der Platte benötigt, wenn man sowohl die PostScript- als auch die Matrix-Fonts verwenden möchte. Im Lieferumfang befinden sich insgesamt 22 Zeichensatzfamilien, die aber nicht alle ganz vollständig sind. So fehlen bei den Effekt-Fonts die kursiven Typen. Für Besitzer von PostScript-Druckern ist natürlich interessant, daß die 11 Schriften, die die LaserWriter von Apple kennen, immer als druckerinterne Zeichensätze erwartet werden. Nur für die übrigen Zeichensätze gibt es Definitionsdateien, die in den Drucker geladen werden. Eine Übersicht über die mitgelieferten Zeichensätze finden Sie in Bild 2.

Die automatische Installation dauert rund 20 Minuten, also genug Muße, erst mal einen Blick auf die einführenden Seiten des Handbuchs zu werfen.

Jetzt geht's los

Der Programmstart von PPM dauert auch seine Zeit. Bis zu dreißig Sekunden vergehen, bis der Startbildschirm erscheint. In dieser Zeit sucht PPM nach einer Datei namens FONTLIST, in der alle zur Verfügung stehenden Zeichensätze aufgeführt sein müssen. Ältere Versionen von PPM haben immer erst nach allen Zeichensätzen gesucht, so daß der Programmstart unhaltbare 5 Minuten in Anspruch nehmen konnte. Die Einrichtung der FONTLIST führt allerdings auch dazu, daß bei Erwerb neuer Fonts diese zunächst im Font-Manager angemeldet werden müssen, bevor sie benutzt werden können. Auch müssen Zeichensätze, die nicht mehr vorhanden sind, abgemeldet werden, sonst hagelt's Bomben. Ist keine FONTLIST vorhanden, wird von Publishing Partner Master eine erstellt. Das dauert dann, je nach Anzahl der Zeichensätze und Speichermedium bis zu 5 Minuten. Überhaupt ist die Verwaltung der Zeichensätze inzwischen ziemlich elegant gelöst. Man kann sich mehrere "Schubladen" (beim ST Ordner) anlegen, in die man die Zeichensätze für verschiedene Anwendungen verfrachtet. So kann man beispielsweise die Zeichensätze für PostScript-Drucker von denen für Matrixdrucker trennen, oder, falls man im Alltagsschriftverkehr nur wenige Zeichensätze benutzt, kann man diese in eine eigene Schublade packen, und nur diese laden lassen. Bei besonderer Gelegenheit kann man eine weitere Schublade aktivieren.

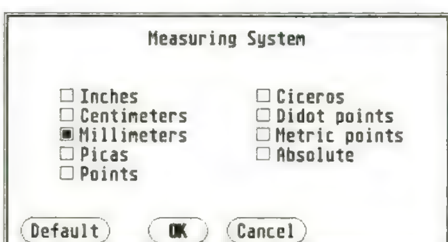


Bild 1: Publishing Partener Master arbeitet mit verschiedenen Maßsystemen.

Bei der Darstellung von Zeichen geht Publishing Partner Master herkömmliche Wege. Während für die Ausgabe auf Drucker Vektorzeichensätze benutzt werden, stehen für die Wiedergabe auf dem Bildschirm die Zeichensätze als Bitmaps in verschiedenen Größen zur Verfügung und werden interpoliert. Dieses Verfahren hat den bekannten Nachteil, daß die Ausgabe auf dem Bildschirm nicht immer die gleiche gute Qualität besitzt, die der spätere Ausdruck hat.

Die Dateien für PostScript-Drucker enthalten entweder Zeiger auf druckerinterne Zeichensätze oder komplette Zeichensätze, die in den Drucker geladen werden können. Im Lieferumfang von Publishing Partner Master finden sich neben Treibern für PostScript-Drucker und Satzmaschinen (Linotronic) auch noch verschiedenste Treiber für Nadel-, Tintenstrahl- und Laserdrucker (ATARI und andere). Um die hohe Auflösung von Satzmaschinen (bis zu 2540 Punkte pro Zoll) erreichen zu können, werden intern alle Linien auf 1/50 Punkt (Setzereinheit) aufgelöst. Ein Punkt entspricht 1/72 Zoll. Damit ein Benutzer sich jedoch nicht unbedingt mit diesen Einheiten auseinandersetzen muß, stehen auf der Anwenderseite die verschiedensten Einheitensysteme zur Verfügung (vgl. Bild 1).

Seitengestaltung

Bevor man sich aufmacht und eine Zeitung oder ein Flugblatt entwirft, sollte man sich zunächst ein grobes Layout aufzeichnen. Man spart dadurch bei der Erstellung des Dokumentes viel Zeit, weil man weiß, was man will.

Zunächst muß entschieden werden, welche Ausmaße eine Seite haben soll. Laut Handbuch sind die Maximalmaße 48 x 48 cm. Falls man jedoch Ausgabedateien für eine Satzmaschine erstellen möchte, gelten andere Grenzen, die von der

Times
Times Bold
Times Ital
Times Ital Bold

Bookman
Bookman Bold
Bookman Ital
Bookman Ital Bold

Helvetica
Helvetica Bold
Helvetica Ital
Helvetica Ital Bold

AvantGarde
AvantGarde Bold
AvantGarde Ital
AvantGarde Ital Bold

✤✻■✳❂✿▼▲ (Dingbats)

Futura XBold
Futura XBold Bold
Futura XBold Ital
Futura XBold Bold Ital

Courier
Courier Bold
Courier Ital
Courier Ital Bold

Helvetica-Narrow
Helvetica-Narrow Bold
Helvetica-Narrow Ital
Helvetica-Narrow Ital Bold

LetteraGoth

Chancery

Oriental

Palatino

Schoolbook
Schoolbook Bold
Schoolbook Ital
Schoolbook Ital Bold

TomHudson

University Roman

Loray Script

Classico (Calsity)

Meteor

Paint Brush

Quadrant

Trellis

Σψμβολ (Symbol)

Bild 2: Die mitgelieferten Zeichensätze

Auflösung abhängig sind. Ein Dokument darf in jedem Fall bis zu 254 Seiten haben, falls der Speicher reicht! Mit Publishing Partner Master lassen sich Dokumente mit einer oder zwei sog. Masterpages erstellen. Auf einer Masterpage kann man Vorgaben machen, die auf jeder Seite im Dokument erscheinen sollen, z.B. die Seitennummer (diese wird automatisch verwaltet). Die Verwendung von Masterpages ist immer dann sinnvoll, wenn man ein mehrseitiges Dokument entwirft, dessen Seiten ein einheitliches Aussehen haben sollen. Bei Postern oder anderen Dokumenten, deren Seiten sich stark unterscheiden, sollte man darauf verzichten. Neben den Masterpages bietet PPM nämlich noch eine ganze Reihe weiterer Möglichkeiten, Seiten sauber zu gestalten. So gibt es ein Raster mit getrennt definierbarer vertikaler und horizontaler Weite, das sich bei Bedarf aktivieren läßt, weiter kann man zusätzliche Führungslinien definieren, an denen man Objekte ausrichten will, und schließlich gibt es die Möglichkeit, verschiedene Objekte aneinander oder an den Seitengrenzen auszurichten (Bild 3).

Text

Das wichtigste Element in einem Dokument ist meistens der Text. Publishing Partner Master kennt zwei Sorten davon. Zunächst ist da die Textspalte, ein Objekt, das mit Text gefüllt werden kann. Dabei ist der Text unabhängig von der Definition der Spalte gestaltbar. Textspalten werden immer dann verwendet, wenn ganze Textabschnitte geschrieben werden müssen.

Neben den Textspalten existieren noch die Textobjekte. Diese verwendet man meistens für Überschriften. Im Gegensatz zu den Textspalten wird bei den -objekten die Größe des Textes verändert, wenn man die Größe des Textobjektes ändert. So kann man leicht die Größe einer Überschrift an eine nachfolgende Textspalte anpassen. Überhaupt ist der Text äußerst vielseitig gestaltbar. So ist es möglich, für ihn unabhängig voneinander Textbreite und -höhe zu definieren. Wenn man Stilattribute ändern will, kann man dies entweder für einen Ausschnitt tun, den man mit der Maus markiert, oder für ganze Textspalten oder das gesamte Dokument. Eine Übersicht der vorhandenen Stilattribute finden Sie in Bild 4. Text in Spalten kann sowohl linksbündig als auch zentriert, als Flatter- oder Blocksatz oder rechtsbündig formatiert werden. Ein- und Ausrückungen sind ebenfalls möglich.

Alle Zeichensätze stehen als Proportionalschrift zur Verfügung. Wenn man Tabulatorsprünge im Text verwendet, werden diese nicht in Leerzeichen umgewandelt. Leider gibt es keine Dezimaltabulatoren, so daß man beim Erstellen von Tabellen viel Zeit benötigt, wenn diese ordentlich aussehen soll. Es ist nämlich möglich, wie auch bei vergleichbaren Produkten, den Zeichenabstand zu ändern. Bei diesem Kerning werden entweder automatisch oder von Hand feste Leerräume zwischen bestimmte Zeichenpaare eingefügt. Durch manuelles Kerning lassen sich (mit viel Zeitaufwand) nicht nur Tabellen ordentlich ausrichten. Normalerweise wird das Kerning automatisch durchgeführt, um ein ansprechendes Textbild zu erreichen. So wird ein 'e' z.B. leicht unter ein 'T' geschoben. Die Zeichenpaare, die gekernt werden sollen, können auch vom Benutzer (um-)definiert werden.

Text wird standardmäßig nach Zeichenabständen ausgerichtet. Falls - besonders bei schmalen Spalten - die Zeichenabstände zu groß werden, kann man auch eine Ausrichtung nach Worten oder eine kombinierte Ausrichtung erzwingen. Falls das Ergebnis immer noch nicht befriedigt, läßt sich der Text mit der halbautomatischen Trennhilfe bearbeiten. Der Testversion lagen nur ein amerikanischer und ein französischer Trennalgorithmus bei, da es sich aber um ein Programmodul handelt, darf man wohl auch mit einer deutschen Version rechnen. Bis dahin sollte man sich mit dem Ausnahmelexikon behelfen können. Allerdings werden die typisch deutschen Besonderheiten dann noch nicht berücksichtigt werden können. Ich denke da u.a. an die Trennung von 'ck' zu 'k-k'.

Wie bisher auch, kann man mehrere Spalten zusammenschalten, so daß Text automatisch von einer Spalte in die gewünschte nächste fließt. Dieses Routing können Sie sich anzeigen lassen, was besonders nützlich ist, wenn man ein so sprunghaftes Layout erzeugt, wie das bekannte Massenblatt mit den vier Buchstaben. Leider gibt es immer noch nicht die Möglichkeit, einen Umbruch in die

Textlayout mit Publishing Partner Professional

In dieser Spalte sehen Sie, wie text direkt bei der Eingabe umgebrochen wird. Es wurde nur das Char-Kerning verwendet. Man kann leicht erkennen, daß so eingegebener text noch der Überarbeitung bedarf, um ein ansprechendes Textbild zu erreichen.

Textlayout mit Publishing Partner Professional

In dieser Spalte sehen Sie, wie Text umgebrochen wird, wenn man Word-Justify verwendet. Auch dieser Text bedarf noch der Überarbeitung.

Textlayout mit Publishing Partner Professional

In dieser Spalte wurde die Kombination von Word-Justify und Char-Justify, das sogenannte Auto-Justify benutzt. Auch hier sind die Ergebnisse nicht 100%ig zufiredenstellend.

Textlayout mit Publishing Partner Professional

In dieser Spalte wurde außer dem Char-Justify zuerst nur der automatische Wortumbruch verwendet. Auch so erhielt man kein akzeptables Ergebnis. Ich hoffe, daß ein deutscher Trennalgorithmus mitgeliefert wird. Nach manueller Trennung sieht der Text übrigens gut aus.

Bild 3: Layoutspielereien

nächste Spalte zu erzwingen, so daß man immer noch selbst Witwen und Waisen (einzelne Zeilen eines Absatzes am Spaltenende oder -anfang) suchen muß. Immerhin gibt das Handbuch Tips, wie man durch geringfügige Manipulationen der Textspalten solche Probleme vermeiden kann.

Für alle die, die trotzdem nicht mit Publishing Partner Master ihre Texte erfassen wollen, gibt es eine umfangreiche Auswahl an Importmodulen für die verschiedensten Textverarbeitungsprogramme. Importiert man Text mit diesen Modulen, werden, soweit das möglich ist, die Textparameter aus der Originaldatei übernommen. Leider wurden aber z.B. beim Importieren der Testseite von 1st_Wordplus die Informationen für die Textbreite nicht übernommen, was darauf deutet, daß die Lineale von 1st_Wordplus (noch?) nicht ausgewertet werden. Neben WordPlus gibt es Importmodule für WordPerfect, Redakteur und ASCII-Dateien.

Grafik

Wie auch schon sein Vorgänger verfügt Publishing Partner Master über einige grafische Fähigkeiten. Zunächst hat der Anwender die Möglichkeit, grafische Objekte, wie wir sie aus EasyDraw und anderen vektororientierten Zeichenprogrammen kennen, in seinem Dokument zu erzeugen. Publishing Partner Master kennt insgesamt 10 Objekttypen: Rechteck, Rechteck mit runden Ecken, vetikale oder horizontale Linie, freie Linie, Kreis, Ellipse, Kreisbogen, Ellipsenbogen, geschlossenes und offenes Polygon und "Freihandzeichnung". Diesen Objekten können jeweils verschiedene Attribute zugeordnet werden. So lassen sich die Farbe, die Liniendicke und -art sowie ein eventuelles Füllmuster frei bestimmen. Konsistent ist das Programm auch beim Freihandzeichnen. Nachdem dort ein Strich gezogen wurde, läßt er sich nur noch als ganzes Objekt ändern, nicht aber mehr die einzelnen Punkte.

Zum Aufpeppen des Layouts sind diese Funktionen allemal ausreichend, auch wenn man ein Dokument etwas aufwendiger gestalten möchte. Neben diesen Zeichenfunktionen besteht aber natürlich auch die Möglichkeit, verschiedenste Grafiken zu importieren. Der Testversion lagen insgesamt 10 Importmodule für verschiedene Grafikarten bei, darunter Encapsulated PostScript (EPS), Metafiles, DEGAS (auch mit Farbe) und IFF.

Publishing Partner Master unterscheidet beim Import von Grafiken zwischen objektorientierten Dateien (Metafiles oder PostScript) und punktorientierten Grafiken. Bei punktorientierten Grafiken besteht die Möglichkeit, die Grafik in ein spezielles Grafikfenster zu laden und dann nur einen Ausschnitt in das Dokument zu übernehmen. Natürlich kann man die importierte Grafik beliebig in der Größe ändern. Aufgrund der hohen internen Auflösung von 3600 Punkten pro Zoll sehen auch stärkere Verzerrungen noch ordentlich aus. Diese Funktion ist somit hervorragend geeignet, aus den überall angebotenen Clip-Art-Bibliotheken Elemente zu übernehmen.

Im Gegensatz zu manchen Grafikprogrammen, interpretiert Publishing Partner Master Metafiles richtig und kann so auch Füllmuster und Linienattribute übernehmen.

Objekt

Alle Elemente, die zur Gestaltung einer Seite dienen, sind Objekte. Jedes Objekt, sei es eine Textspalte, ein Textobjekt, eine Grafik oder nur eine Linie, kann einzeln selektiert und verändert werden. Eine besondere Rolle spielen dabei die Textspalten. Bei diesen handelt es sich immer um Rechtecke, die mit einem weißen Hintergrund gefüllt sind. Es ist aber möglich, diese Einstellung zu ändern, und so Durchscheineffekte zu erzeugen. Die Attributeinstellungen für die Textspalte (als Objekt) sind aber unabhängig von den Einstellungen für den Text in der Spalte. Am Anfang ist dies ein wenig verwirrend, aber wenn man das Prinzip mal verstanden hat, winkt die völlige gestalterische Freiheit.

Für alle Objekte gilt: sie lassen sich zu beliebigen Gruppen zusammenfassen, man kann Objekte übereinander stapeln, und es ist möglich, Objekte oder Gruppen gegen Veränderungen zu schützen.

Mit diesem Wissen ist Publishing Partner auch für andere Dinge zu gebrauchen, als nur für die Erstellung von Briefbögen. Ich habe mir einen Grundriß meiner Wohnung gemacht und ummöbliert.

Hilfe bei der Arbeit

Dabei kamen mir einige Aspekte von Publishing Partner Master besonders zugute. Das gesamte Programm (seltsamerweise mit Ausnahme des Font-Managers) ist vollständig über die Tastatur steuerbar, auch die Knöpfe in den Dialogfeldern. So kann man für die wichtigsten, immer wiederkehrenden Arbeitsschritte die zehn Funktionstasten mit Makros belegen.

Ein weiterer Aspekt, den einige Leser vielleicht von anderen Programmen her kennen, sind die Absatz- und Seitenformate. Mit den Absatzformaten (engl. tags) kann man sämtliche relevanten Attribute für Textabschnitte festlegen. So ist es z.B. möglich, für Überschriften eine Schmuckschrift in der Größe 36 Punkte zu wählen, doppelt unterstrichen, linksbündig, einfach gesperrt, drei Punkte Zeilensperrung, die Buchstaben mit einem Füllmuster gefüllt und das ganze in Rot. Wenn wir uns ein solches Absatzformat mit dem Namen 'ÜBER' erstellen, brauchen wir später in einem Text nur noch einen Abschnitt zu markieren und dieses Format auszuwählen. Mit Absatzformaten ist es ein leichtes, auch längeren Texten ein einheitliches und doch vielfältiges Aussehen zu geben, ohne zig Parameter im Kopf haben zu müssen. In einem Dokument dürfen bis zu 255 verschiedene Absatzformate definiert werden.

Eine große Hilfe sind auch die vielen Formate, in denen PPM eine Seite auf dem Bildschirm darstellen kann. Mit Hilfe des variablen Zooms kann man notfalls auch eine beliebige Vergößerung zwischen 15% und 1500% einstellen. Die Darstellungsqualität gegenüber den Vorgängerversionen hat sich zumindest bei den mitgelieferten Zeichensätzen erheblich verbessert, was auf die größere Zahl verfügbarer Bildschirm-Fonts zurückzuführen ist. Eine letzte, leider nicht ganz gelungene, Hilfe ist die UNDO-Funktion. Nicht ganz gelungen ist sie, weil sich nicht alle Operationen zurücknehmen lassen. Da kann schon mal Frust aufkommen, wenn man im Objektmodus mal aus Versehen die <Delete>-Taste drückt und eine sorgfältig gestaltete Spalte sang- und klanglos ins Datennirwana entschwindet.

Viel Verwaltungsarbeit

An dieser Stelle ist es vielleicht angebracht, etwas über die Verwaltung der Attribute im Text zu sagen. Publishing Partner legt bei Anwahl eines neuen Attributes ein unsichtbares Zeichen im Text ab, so daß der gesamte nachfolgende Text dieses neue Attribut erhält. Wurde ein Block markiert, wird am Blockende ein weiteres unsichtbares Zeichen eingesetzt, das das Attribut wieder löscht. Genauso ist es mit Absatzformaten, nur daß in einem Absatzformat viele Attribute umdefiniert werden können. Löscht man nachträglich Text, werden auch die darin enthaltenen Attributzeichen und Formatzeichen entfernt, so daß nachfolgender Text eventuell ein völlig anderes Aussehen erhält. So läßt sich auch eine Funktion von Publishing Partner Master erklären, die eigentlich

Rückenlage
fett
doppelt unterstrichen
kursiv (pseudo)
hell
Spiegelschrift
umrandet
invers
Schattenschrift
durchgestrichen
unterstrichen
Kopfüber

Bild 4: Hier werden die Textattribute demonstriert.

selbstverständlich ist, aber in der Urversion noch nicht existierte: das automatische Anpassen von Attributen. Wenn man in bestehende Textelemente Text einfügt, erhält dieser immer die gleichen Attribute wie der letzte vorstehende Buchstabe, auch wenn in den Menüs andere Parameter eingestellt sind.

Bonbons

Neben dem bisher Aufgezählten bietet Publishing Partner Master noch drei weitere neue Möglichkeiten, die helfen, das Layout interessant zu gestalten. Zunächst ist da der automatische Textfluß um Objekte. Um jedes geschlossene, beliebig geformte Objekt kann man Text herumfließen lassen. Falls das gewünschte Objekt keine geschlossene Umrißlinie hat, z.B. bei punktorientierten Grafiken, kann man mit dem Polygonzug schnell eine unsichtbare Linie erzeugen. Es empfiehlt sich allerdings, zunächst einmal mit dieser Option ein wenig zu experimentieren, bevor man sie in eigenen Dokumenten einsetzt. Die Beschreibung im Handbuch läßt ein wenig zu wünschen übrig, der amerikanische Hersteller hat jedoch einen Newsletter herausgebracht, der diesen Mangel behebt. Bleibt zu hoffen, daß er auch in Europa zu bekommen sein wird.

Die zweite Option, die sich sehr vorteilhaft einsetzen läßt, ist die Möglichkeit, Text in Gradwinkeln rotiert in den Text einzufügen. Dabei kennt Publishing Partner die Möglichkeit, nur die Grundlinie des Textes oder auch die Buchstaben selbst zu drehen. Die Winkelangaben entsprechen denen des aus der Schule bekannten xy-Koordinatensystems. Alle Eingaben zur Drehung von Text werden nach Abschluß der Rotation so optimiert, daß immer möglichst kleine Winkel im Dialogfeld stehen.

Farbe

Publishing Partner ist in der Lage, auch farbige Dokumente zu erstellen. Eine Einschränkung sind dabei die vom Computer darstellbaren Farben. Mit dem monochromen Monitor bringt es der ATARI nur auf zwei, der AMIGA, auf dem das Programm auch läuft, ist da wesentlich besser ausgestattet. Publishing Partner kann daher den Farben, die später gedruckt werden sollen, eine Bildschirmfarbe zuordnen, auch verschiedenen Farben die gleiche Bildschirmfarbe. Über die Anzahl von Farben, die Publishing Partner auf einmal verwalten kann, schweigt sich das Handbuch leider aus. Da aber theoretisch insgesamt über 282 Millionen Farbtöne erzeugt werden können, dürften es eine ganze Menge sein.

Doch wie wird die Farbe erzeugt? Wer in seinem Kopf mal ganz unten kramt und sich an den Kunstunterricht oder Physik erinnert, weiß noch, daß man Farben durch Mischung erzeugt. Dazu kennt man die additive Farbmischung, wie bei einem Lichtstrahl, bei dem aus den Primärfarben Rot, Grün und Blau als Produkt Weiß entsteht. Mischt man die Primärfarben in verschiedenen Verhältnissen, so erhält man die Sekundärfarben Türkis, Violett und Gelb. Diese Sekundärfarben sind die Primärfarben der zweiten Form der Farbmischung, der subtraktiven. Die subtraktive Farbmischung spielt in unserem Alltagsleben eine noch größere Rolle als die additive, weil sie unser Farbsehen immer bestimmt, wenn ein Objekt (diese Seite z.B.) Licht reflektiert. Werden die Primärfarben bei der subtraktiven Farbmischung gemischt, entsteht als Summe Schwarz, weil kein Licht mehr reflektiert wird.

Publishing Partner kennt noch drei weitere Farbmischsysteme, darunter auch das der amerikanischen Fernsehnorm NTSC. Man unterscheidet darüber hinaus noch zwei Farbtypen: Farben, die nicht durch Farbmischung entstehen können, wie Metallic-Effekte oder Gold und Silber, und trennbare Farben, die sich mit einem der erwähnten Verfahren aufteilen lassen.

Und auf dem Papier?

Die Qualität eines Desktop-Publishing-Programmes steht und fällt mit dem Druckergebnis. Dieser Tatsache ist man sich bei SoftLogik und Upgrade offensichtlich bewußt, hat sich doch bei der Qualität der Ausdrucke viel getan. Diejenigen von Ihnen, die noch die Aprilausgabe der ST-Computer von 1987 besitzen, können ja mal einen Blick auf die Seiten 111 und 113 werfen. Der wichtigste Punkt bei der Verbesserung der Druckqualität war of-

fensichtlich eine Überarbeitung der mitgelieferten Zeichensätze, insbesondere hinsichtlich des Kernings durch PPM. Die Ausgabe auf dem 24-Nadeldrucker konnte durchaus überzeugen, auch wenn ein Vergleich mit dem Ausdruck des Laserdruckers natürlich hinkt. Allerdings wurden auch die druckereigenen Zeichensätze verwendet. Leider war der Treiber für die LQ-Drucker von Epson noch nicht in der Lage, die höchste Auflösung von 360*360 Punkten pro Zoll zu nutzen.

Die Dateien für den PostScript-Drucker wurden übrigens zunächst auf Diskette geschrieben, auf eine 5 1/4-Zoll-Diskette im IBM-Format gearct und dann bei einem nahegelegenen Computerladen auf einem PC wieder ausgepackt und schließlich an den Drucker geCOPY't. Das lief wie am Schnürchen, jedenfalls solange nicht zuviele ladbare Zeichensätze auf einmal in den Drucker geschickt wurden. In diesem Fall traten Probleme auf. Probeweise habe ich auch eine Vierfarbtrennung von einem Dokument mit bunten Bildern gemacht. Beim Ausdruck von Farbbildern gibt es verschiedene Möglichkeiten. Man kann entweder für jede Farbe einen einzelnen Druckdurchgang oder für die separierbaren Farben zusammen eine CMYK-Trennung machen und nicht separierbare Farben auf je einer zusätzlichen Seite ausdrucken. Für die Besitzer von echten Farbdruckern besteht auch die Möglichkeit, ohne Farbtrennung zu drucken.

Zeitweise wolkig

Bei so viel guten Eigenschaften des Programmes muß auch irgendwo ein Pferdefuß sein, denn fehlerfreie Software gibt es bekanntlich nicht. Auch Publishing Partner Master ist nicht die rühmliche Ausnahme. Zwar gab es während des Tests selber nur einen Absturz, aber der ließ sich nicht einwandfrei reproduzieren, was ihn so besonders unangenehm macht. Es fielen dabei auch keine Bomben, aber selbst nach 45 Minuten war der Rechner zu keiner Reaktion zu bewegen. Das Grundproblem scheint mir bei diesem Absturz die Verwaltung des Speichers zu sein. Wie wir gesehen haben, ist es mit PPM möglich, auch Textobjekte zu drehen. Diese behalten aber auch bei einer Drehung um 90 Grad ihre ursprüngliche Breite bei, da sie um den Mittelpunkt des Objekts gedreht werden. Leider ist das auf dem Bildschirm nicht zu sehen, und so kann es passieren, daß man ein Objekt zu nah an den Seitenrand rückt, wo es dann plötzlich verschwindet und nicht wieder sichtbar gemacht werden kann. Aus diesem Grund wird im Handbuch auch empfohlen, solche Objekte durch Einfügen von Leerzeilen vor dem Drehen quadratisch zu machen. Trotzdem sind mir einige Objekte auf diese Art abhanden gekommen, bis sich irgendwann nichts mehr tat.

Die andere Unart des Programms war der Absturz, der bei Verwendung des LQ-Treibers regelmäßig nach dem Ausdrukken der ersten Seite auftrat. Dieses Phänomen ließ sich auch auf einem eilends herbeigeschafften, zweiten Rechner repoduzieren, schieben wir's also auf den Drukkertreiber. Hier sollte nochmals eine sorgfältige Überprüfung stattfinden. Dieses Problem entstand übrigens immer nur mit dem EPSON-LQ-Treiber, nicht bei der Verwendung des PostScript- und des ATARI-Laser-Treibers.

Na und?

Sie haben sich also bis hierher durchgekämpft - nein? Dann aber schnell zurück an den Anfang - und wollen wissen, wieviel Sie für solch ein Stück Software anlegen müssen? Die Antwort lautet: 798.- DM. Dieses Geld ist aber bestimmt gut angelegt, erhält man doch ein weitgehend ausgereiftes Produkt, das - dies sei besonders betont - wirklich einfach zu bedienen ist, wenn man auch am Anfang wie bei jedem Programm eine gewisse Lehrzeit einkalkulieren muß. Die Philosophie des Programmes ist durchgängig, wenn auch nicht ganz eng an den Maßstäben der Zunft der Setzer angelegt. Denen würde es nie einfallen, einen Zeichensatz einfach beliebig zu verzerren; wo bleiben denn da die gefälligen Proportionen?

Für den Anwender mit gehobenen Ansprüchen ist jedoch ein (PostScript-) Laserdrucker unverzichtbar, vielleicht tut es ja auch der Laserdrucker von ATARI, wenn er endlich einmal ULTRASCRIPT (ein PostScript-Clone) versteht. Die vorliegende Version des Programmes kann man jedenfalls empfehlen, im Gegensatz zu vielen Vorgängern, die sehr viel stärker absturzgefährdet waren.

Ausblick und Dank

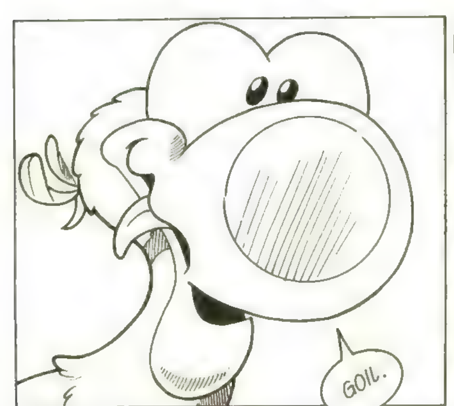
Nachforschungen jenseits des großen Teiches ergaben übrigens interessante Neuigkeiten. Für Ende des Jahres ist die Version 2.0 von Publishing Partner geplant, dann mit Vektorzeichensätzen von CompuGraphic und anderen namhaften Zeichensatzmachern auch für den Bildschirm. Falls der deutsche Vertrieb die gute Update-Praxis des Mutterhauses übernimmt, braucht man aber trotzdem nicht zu warten.

CSM

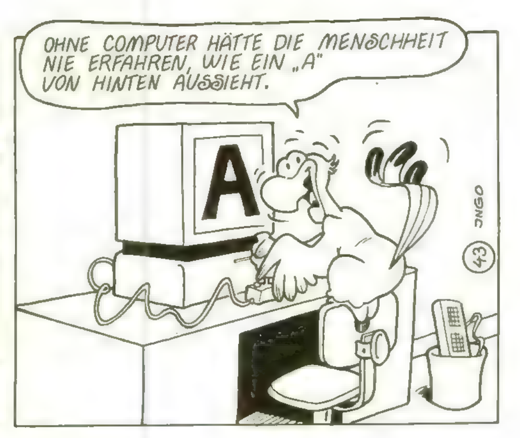
Bezugsadresse:

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt
Tel.: 06151/56057







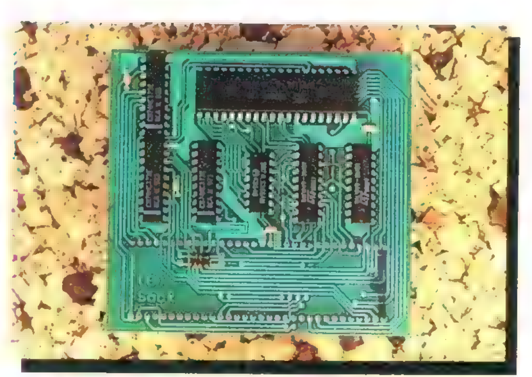


Die *MS-DOS*-Welt erschließen mit PC-Speed ...

PC Speed für den ST

Über 24000 Anwender haben weltweit in den ersten neun Monaten nach seiner Einführung PC-Speed gekauft und sich mit diesem Spitzenprodukt Zugang zu den Tausenden von MS-DOS Programmen verschafft.

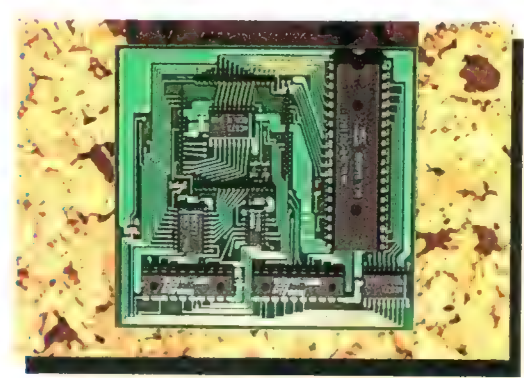
Neuer Verkaufspreis 398,- DM (Unverbindlich empfohlener Verkaufspreis)



PC-Speed erweitert den ATARI ST zum MS-DOS-Rechner

PC Speed für den STE

Obwohl PC-Speed für den STE erst seit Wochen auf dem Markt ist, haben schon über 2000 Benutzer zugegriffen und integrieren MS-DOS Programme wie selbstverständlich in ihre Computerarbeit.



PC-Speed erweitert den ATARI STE zum MS-DOS-Rechner

Was hat PC-Speed diesen Erfolg gebracht?

- die modernste Technologie
- die saubere Lösung
- die sichere Funktion
- die hohe Leistung
- die High-Tech-Qualität
- die Verfügbarkeit im Bundesgebiet bei über 400 Fachhändlern
- die gute Hotline-Betreuung
- der Update-Service

Mit PC-Speed profitieren ST- und STE-Anwender von der Leistung ihres ATARI-Computers und gleichzeitig von der unübertroffenen Programm-Vielfalt der MS-DOS Welt

M:OOP
Magic OOP?

Während der PC-Markt mit objektorientierten Programmiersprachen überflutet wird, hat der ST-Anwender nur wenige Möglichkeiten, sich mit diesem neuen Programmierparadigma auseinanderzusetzen. M:OOP, ein Objective-C-kompatibler Precompiler, verspricht Abhilfe.

Da sicherlich nur die wenigsten etwas über Objective-C wissen, gliedert sich der Testbericht in zwei Teile: eine kurze Übersicht der Sprache und der OOP (objektorientierte Programmierung) und den Test selber.

Objective-C

In seinem Buch [1], beschreibt Cox einen Weg, wie man die objektorientierten Elemente von Smalltalk-80 mit der Ausdrucksstärke von C verbinden kann. Das Ergebnis ist eine Sprache, die leider sehr wenig nach C aussieht, dafür aber sehr einfach zu lernen und zu implementieren ist - vorausgesetzt man kann bereits C und hat auch einen entsprechenden Compiler. Diese neue Sprache soll, wie auch OOP im allgemeinen, große Programmierprojekte erleichtern und Software wiederverwendbar machen. Cox spricht dabei von dem Software-IC-Modell. Wie die aus der Elektronik bekannten Chips, sollen Programmteile, die in sich völlig abgeschlossen sind, in große Programme eingesetzt werden. Damit wird verhindert, daß 'Software-Räder' zum hundertsten Mal neu erfunden werden, und man stellt sicher, daß Teile von Programmen, hier Objekte genannt, fehlerfrei sind. Objekte sind von C her bekannte Strukturen, die zusammen mit den Funktionen, die auf sie zugreifen (Methoden genannt), innerhalb eines Moduls, d.h. einer Datei definiert werden. Ausgehend von dieser Veränderung, erleichtern einige Begriffe aus der OOP-Literatur dem Programmierer das Arbeiten am Computer.

Data Encapsulation

Die in den Objekten definierten Variablen können von außen her nicht verändert werden, wohl aber durch die Methoden, die im selben Objekt zu finden sind. Das wirkt sich sehr positiv auf den Programmierstil aus, denn dadurch werden globale Variablen vermieden und das Programm spaltet sich zwangsweise in kleine, überschaubare Teile. Vorteile hat die Verkapselung der Daten auch in sehr großen Projekten, wenn die Gefahr besteht, daß sich Namen in verschiedenen Modulen gleichen.

Single Inheritance

Eine der besonderen Stärken von OOP ist die Vererbung: haben Sie zwei oder mehrere Objekte, die sich in einigen Eigenschaften gleichen, so definieren Sie erst eine Klasse (so heißen Objekte) mit genau diesen Eigenschaften und vererben diese dann weiter auf Ihre eigenen Objekte. Die Vorteile liegen auf der Hand: Redundanz entfällt, und wenn Sie ein Objekt ergänzen, so ergänzen Sie seine Nachfolger mit. In Objective-C ist die Basisklasse das 'Object', ein Objekt mit der minimalen Funktionalität, d.h. es enthält nur das Notwendigste. Alle von Ihnen definierten Objekte erben Variablen und Methoden von dieser Klasse. Sie können nun zusätzliche Variablen und Methoden definieren, bestehende überlagern oder löschen. Im Gegensatz zur multiplen Vererbung können hier die Objekte nur einen Vorgänger haben. Dies erhöht die Übersicht jedoch erheblich.

Polymorphism

Zusätzlich zu den in C bereits bestehenden Typen wird in Objective-C ein neuer eingeführt: 'id'. Dieser Typ kann alles mögliche sein: vom Integer bis zu ganzen Objekten. Mit Hilfe dieses Typs können die sogenannten generischen Funktionen geschrieben werden. Ein Beispiel dazu: Sie schreiben eine Sortierroutine, die Elemente vom Typ 'id' sortiert. Es ist Ihnen dabei völlig egal, was 'id' nun wirklich ist; Hauptsache, es hat eine Vergleichsmethode, mit der Sie entscheiden können, welches von zwei Objekten größer ist. Nun können Sie mit Ihrer Routine Integer-Arrays, Namen und beliebige andere Objekte sortieren. Die gesamte Verwaltung übernimmt die Laufzeitumgebung des Objective-C.

Late Binding...

...ist eines der stärksten Features von OOP. Der Prozeß des Linkens wird hierbei auf die Laufzeit des Programms aufgeschoben. Erst so wird Typunabhängigkeit möglich. Eine weitere Anwendung findet sich in sogenannten Collections, einer vordefinierten Klasse. Dieser Klasse können Sie Objekte zuweisen, die zum Beispiel alle eine gemeinsame Methode haben. Wenn Sie nun diese Methode über eine besondere Funktion der Collection als Parameter übergeben, wird sie jedes darin enthaltene Objekt ausführen. Insbesondere eignet sich diese Eigenschaft zur Bewältigung von Aufgaben, die mit der grafischen Oberfläche zu tun haben.

M:OOP

Alle oben aufgeführten 'Features' werden auch von M:OOP, einem Produkt der Berliner Firma Fries und Partner, unterstützt. Neben der Programmdiskette befinden sich in der Verpackung noch ein 85 Seiten starkes Handbuch und ein Lizenzvertrag. Da M:OOP nur ein Precompiler ist und C-Code generiert, braucht man auch einen C-Compiler. Die Version 2.0 unterstützt Mark-Williams-C und Turbo C. Neben den beiden Anpassungen befinden sich auf der Diskette die Sourcecodes für die acht Standardklassen und für ein größeres Programmbeispiel, eine Bibliothek, die die Standardklassen und die Laufzeitumgebung enthält, und, wie sollte es anders sein, der M:OOP-Compiler selbst. Dieser wird in zwei Ausführungen geliefert: einer Commandline- und einer Accessory-Version (Bild 1).

Das Handuch geht in der ersten Hälfte auf das Programmieren in M:OOP ein. In diesem Teil werden auch die mitgelieferten Klassen detailliert erklärt. In der zweiten Hälfte ist das größere Beispiel abgedruckt, das man auch auf der Diskette findet. Der Index des Buches ist erstaunlich gut ausgefallen: allewichtigen Begriffe sind aufgeführt.

Der Compiler...

... wird, wie oben schon erwähnt, in zwei Versionen geliefert. Wenn Sie öfters unter einer Text-Shell arbeiten, werden Sie keine Probleme beim Integrieren des Compilers in Ihre Programmierumgebung haben. Die einzige Änderung: Bevor die Source-Dateien an den C-Compiler kommen, werden sie an M:OOP geschickt. Jede Klasse, die Sie neu definieren, wird in einem besonderen Ordner in Form einer '.DAT'-Datei gespeichert. Diese entspricht ungefähr den üblichen Header-Dateien und muß in die Module, die diese Klasse benutzen, eingebunden werden. Der Ordner muß dem Compiler mitgeteilt werden, indem eine entsprechende Environment-Variable gesetzt wird. Hier treten die ersten Probleme mit der Accessory-Version auf. Es gibt hier keine Möglichkeit, diese Variable zu setzen. Der Compiler sucht seinen Ordner dann auf dem aktuellen Laufwerk, wo er sich mit größter Wahrscheinlichkeit nicht befindet. Die zu übersetzenden Dateien müssen manuell per File-Selector bestimmt werden. Das kann besonders bei größeren Projekten lästig werden. Ein weiteres Manko: Obwohl das Accessory nur um die 50kB lang ist, verbraucht es im Speicher ca. 250kB. Hier kann es einem leicht passieren, daß man den C-Compiler wegen Speichermangel nicht mehr laden kann.

Was die Geschwindigkeit angeht, liegt M:OOP hinter Turbo C. Beim Übersetzen der Klasse 'Set' ergaben sich folgende Werte:

M:OOP: 271 Zeilen in 8030 ms
Turbo C: 401 Zeilen in 6140 ms

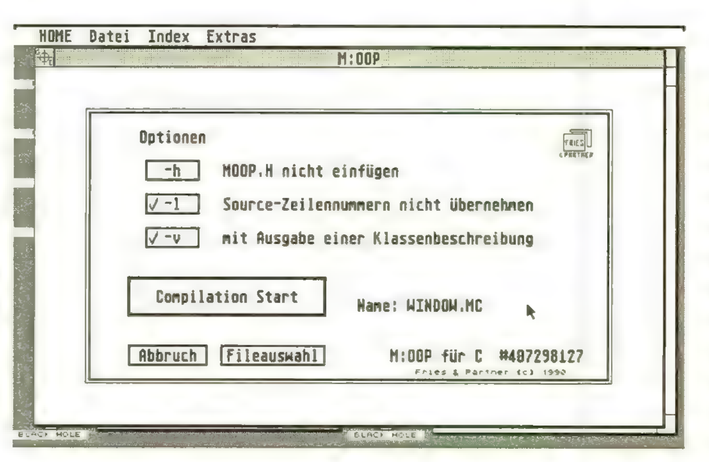
Der erzeugte Code ist kompakt und stützt sich auf die Laufzeitumgebung. Diese besteht im wesentlichen aus der Funktion *_msg*. Ihr Programm erhält dadurch einen Overhead von nur etwas über 2kB (plus die benötigten Klassen). Jede von Ihnen definierte Methode bekommt ihre eigene Nummer. Diese wird dann an *_msg* weitergegeben und intern auf die richtige Adresse umgewandelt. Die dabei entstehenden Verzögerungen sind je nach der Programmgröße unterschiedlich, aber durchaus als gering zu bezeichnen: eine leere Methode ist ungefähr zehnmal langsamer als eine leere Funktion in Turbo C. Bei Methoden mit 'Inhalt' fällt dies jedoch nicht ins Gewicht.

Sechs im Handbuch beschriebene globale Variablen bilden eine sehr sinnvolle Möglichkeit, den Dispatcher zu beobachten und zu erweitern. So können alle Methodenaufrufe protokolliert, die Methodentabelle erweitert oder neue objektorientierte Speicherverwaltungsroutinen integriert werden.

Die Fehlermeldungen sind zwar nicht in Unmengen vorhanden, dafür aber deutsch. Leider untersucht M:OOP die Quelle nicht besonders gründlich nach Fehlern, so daß sie erst beim Compilieren mit dem C-Compiler zutage treten. Das verzögert die Arbeit beträchtlich.

Wie oben schon erwähnt, bauen alle Klassen in Objective-C auf einer Standardklasse, nämlich 'Object', auf. Diese und sieben andere befinden sich im Quellcode auf der Diskette. Sie stellen das Fundament der Programmierung in M:OOP dar. So sind zum Beispiel einige Array- und Collection-Klassen enthalten, die man als Anwender nicht implementieren könnte. Bevor man sie benutzen kann, muß man sie erst einmal übersetzen. Dies ist um so lästiger, da diese bereits in übersetzter Form in der Bibliothek enthalten sind.

Was die Kompatibilität zu Objective-C von Stepstone angeht, so muß gesagt werden, daß M:OOP die '@-Direktiven' nicht kennt. Daher ist eine Portierung von M:OOP auf Objective-C leichter zu bewerkstelligen als umgekehrt. Alle anderen Inkompatibilitäten sind eher nebensächlich und im Handbuch beschrieben.



Fazit

Mit 198 DM können Sie nun endlich anfangen, objektorientiert zu programmieren. Die Frage ist nun sicherlich, für wen sich dieses Produkt eignet. Einerseits ist der Preis etwas hoch ausgefallen, vor allem wenn man bedenkt, daß man für den C-Compiler extra zahlen muß. Andererseits ist der Lieferumfang ziemlich spartanisch: ein Browser und Profiler sollten in einem professionellem Paket schon enthalten sein. Und dann ist da ja noch die Accessory-Version; sie stellt eine Art Verlegenheitslösung dar. Eine schöne GEM-Shell mit einer Make-Funktion würde dem Programm gut zu Gesicht stehen. Das Handbuch stellt zwar keinen Ersatz für [1] dar, läßt sich aber trotz orthographischer Fehler gut lesen und bildet für den Anfang eine solide Grundlage. Wenn der Hersteller sein Versprechen, die Programmierer mit Klassen zu unterstützen, hält, dann kann ich das Programm jedem empfehlen, der unbedingt die Vorzüge von OOP kennenlernen will. Der Weisheit letzter Schluß ist M:OOP jedoch nicht.

Grischa Ekart

Bezugsquelle:

Fries & Partner
Eislebener Str. 7
1000 Berlin 30
Tel.: 030/69 41 114

Literatur:

[1] B.J. Cox: Object-Oriented Programming Addison-Wesley, 1986

fastGEM

GEM-Bibliothek für GFA-Assembler



Welcher begeisterte Assembler-Programmierer stand nicht schon vor der Situation: Nach ersten Begegnungen mit den Low-Level-Funktionen des Betriebssystems kommt der Wunsch auf, außer der Geschwindigkeit auch etwas Bedienerfreundlichkeit zu bieten - mithin die eigenen Programme optisch aufzuwerten.

Neidisch blickt man den Besitzern höherer Programmiersprachen über die Schulter, die ohne großen Aufwand von der grafischen Benutzeroberfläche des ST Gebrauch machen. Während deren Werke mit Menüleisten, schattenwerfenden Fenstern und Auswahlboxen glänzen, bleiben dem Assembler-Anfänger meist nur die geringen Möglichkeiten des TOS. Dieses beschränkt sich bei der Bildschirmausgabe mittels GEMDOS-Routinen und Escape-Sequenzen auf die Darstellung von Texten im Zeichenraster, eventuell noch invers hervorgehoben.

Doch das für die erwünschte Grafik zuständige GEM hat eine etwas kompliziertere Aufrufstruktur, die in den meisten Abhandlungen nicht ausreichend beschrieben wird. Entweder ist nur die Implementation in C aufgeführt, oder es wird der Ratschlag gegeben, die gewünschten Befehle in Pascal, Modula oder C zu programmieren und dann die Maschinensprache einzubinden.

Aufruf der GEM-Funktionen

Hat man sich trotzdem einmal durch das Literaturdickicht zu einer brauchbaren Beschreibung durchgearbeitet, stellt sich eine ernüchternde Situation dar.

Im Unterschied zu den TOS-Aufrufen werden die Parameter bei den GEM-Funktionen nicht auf dem Stack übergeben, sondern in globalen Feldern abgelegt, die vom Programm definiert sein müssen. Hinzu kommt, daß man auch die Anzahl der Übergabe- und Rückgabeparameter explizit angeben muß. So artet ein Aufruf in viel Fleißarbeit aus.

Abhilfe

Das soll sich nun mit fastGEM von der Darmstädter Firma S&J ändern. fastGEM ist eine GEM-Bibliothek für den GFA-Assembler und umfaßt 124 VDI- und 71 AES-Aufrufe. Die zwei neuen AES-Routinen *fsel_exinput* und *wind_new* ab TOS 1.4 sind ebenfalls enthalten.

Positiv fällt der äußere Eindruck des Produktes aus: fastGEM wird mit einem stabilen Kunststoffordner geliefert, der die ca. 400 Seiten starke Anleitung beinhaltet. Auch an Kleinigkeiten wie die extra beigefügten Diskettenaufkleber für die Sicherheitskopie wurde gedacht.

Auf der Diskette befinden sich der Sourcecode der Library sowohl in ASCII als auch in tokenisierter Form für den GFA-Assembler. Einige beigefügte Demonstrationsprogramme runden das Paket ab. Die Beispiele sind recht einfach gehalten und erlauben einen ersten Einblick in die neuen Programmiermöglichkeiten.

Die Library

Sämtliche GEM-Funktionen in der 109 kB langen Library sind als Makros definiert. Innerhalb jedes Makros werden die Anzahl der Variablen automatisch gesetzt, die Übergabewerte in den einzelnen Feldern eingetragen und die betreffende Betriebssystemroutine ausgeführt.

Der Autor von fastGEM hat sich dabei größtenteils an die C-Syntax gehalten, so daß die Funktionsnamen und Bezeichnungen der Parameter mit denen eines C-Aufrufs übereinstimmen. Jedoch werden keine Rückgabewerte (außer bei *graf_handle*) unterstützt und folglich bei einem Makroaufruf auch nicht angegeben. Dieser Umstand macht es für den Programmierer erforderlich, eventuelle Rückgabeparameter selbst aus den entsprechenden Feldern herauszuholen.



Das Ergebnis des Beispielprogramms auf dem Monitor

Neben den Makros für die einzelnen Funktionsaufrufe sind noch eine Vielzahl von numerischen Werten wie z.B. Fenster-Flags, Objekt- und Resource-Attribute sowie die globalen Felder für VDI und AES vordefiniert. Es empfiehlt sich daher, den Sourcecode der Library gut durchzusehen, um unbeabsichtigte Doppeldeklarationen zu vermeiden.

Das Handbuch widmet jeder Funktion eine eigene Seite. Diese enthält eine kurze Erklärung der Routine, die Syntax des Makro-Aufrufs und die Aufzählung der Rückgabeparameter. Außerdem ist für interessierte Benutzer noch die Funktionsdeklaration in Assembler angegeben. Auf den freigelassenen Rückseiten hat man mehr als ausreichend Platz, seine Erfahrungen mit den Betriebssystemroutinen festzuhalten.

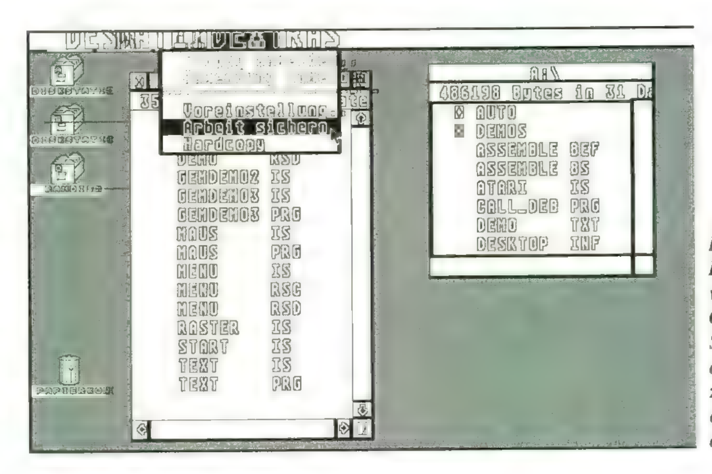
Für eine detailliertere Abhandlung der Funktionen, insbesondere über die Bedeutung der Parameter, wird auf GEM-Literatur verwiesen. Denn auch hier gilt, daß die mitgelieferte Beschreibung keinen GEM-Kurs ersetzt.

Aus diesem Grund wird fastGEM zusammen mit dem Atari ST Profibuch und dem GFA-Assembler als Komplettpaket angeboten. Es besteht aber auch die Möglichkeit, fastGEM alleine zu erwerben.

Test

In der Praxis lassen sich mit fastgem schnell übersichtliche Quelltexte für GEM-Programme erzeugen. Das abgedruckte Demonstrationsprogramm konnte z.B. in 20 Minuten geschrieben werden. Die notwendigen Vorbereitungen sind dabei minimal: Mit der Assembler-Direktive INCLUDE wird die Library am Anfang des eigenen Programms eingebunden, und schon stehen die GEM-Funktionen zur Verfügung. Zum Aufrufen gibt man einfach ihren Namen und die relevanten Parameter an. In diesem Zusammenhang stört nur, daß man zwischen Handbuch und Begleitliteratur wechseln muß, um einerseits die korrekte Syntax und andererseits die Parameterbedeutung in Erfahrung zu bringen. Laut Hersteller ist deswegen eine kartonierte Beilage in Vorbereitung, die eine Kurzübersicht der Funktionsaufrufe enthalten wird.

Wie bei allen Makroanwendungen bezahlt man den gewonnenen Komfort mit einem Verlust an Kompaktheit: Öfters verwendete Funktionen treten im fertigen Programm ebenfalls mehrfach als Code



Recht abenteuerlich ist die Rückkehr zum Desktop, wenn geänderte Grafikparameter (hier Schriftart und Linienstärke aus dem Beispiel) nicht zurückgesetzt wurden. Bei einigen Accessories führt das sogar zum Absturz.

auf. Speicherplatzfetischisten dürfte es allerdings keine Schwierigkeiten bereiten, die Makros in Unterprogramme umzuändern. Ihnen entgeht dann aber die größere Flexibilität der Konstruktion.

Tips

Dem Handbuch zufolge sollen die meisten Parameter als Direktwerte übergeben werden. Es ist jedoch ohne weiteres möglich, stattdessen Datenregister zu verwenden (siehe Listing). Da der größte Teil der Parameter in den Makros per MOVE-Befehl in die globalen Felder übertragen wird, sind sogar praktisch alle Adressierungsarten bei der Übergabe erlaubt! Ausnahmen bilden Felder und die Variable *handle*, die über ihren Label-Namen angesprochen werden müssen.

Diese undokumentierte Eigenschaft kann bei entsprechender Programmierung etliche Maschinenbefehle einsparen und gleicht so den oben angesprochenen Nachteil wieder aus. Trotz der mächtigen Befehle muß der Assembler-Programmierer größere Sorgfalt walten lassen als bei anderen Programmiersprachen: Eine korrekte An- und Abmeldung der Applikation beim Betriebssystem ist zwar in einem Beispiel vorgegeben. Damit werden allerdings gewisse Parametereinstellungen nicht rückgängig gemacht, wie man es von den höheren Sprachen her gewohnt ist. So bleiben Änderungen von Schriftart/-größe und Linienart nach Programmbeendigung bestehen. Die Rücksetzung auf die ursprünglichen Werte muß vom Programmierer vor der Applikationsabmeldung durchgeführt werden.

Grundsätzlich kann man einen anderen als den GFA-Assembler verwenden; es sind jedoch einige Voraussetzungen zu erfüllen. Der Assembler muß mindestens 16 Übergabewerte für Makros unterstützen und die Markierung "\ ~" erkennen, die lokalen Labels innerhalb eines Makros vorangestellt wird.

Fazit

fastGEM ist eine leistungsfähige Library für Besitzer des GFA-Assemblers, die ohne großen Aufwand auf die umfangreichen Grafikroutinen des ATARI ST zugreifen wollen. Anfängern mit geringen GEM-Kenntnissen sei die empfohlene Literatur unbedingt ans Herz gelegt.

Preise:

fastGEM, GFA-Assembler und Atari ST Profibuch: 369,-
fastGEM: 179,-

Bezugsquelle:

S & J
Otto-Röhm-Straße 81
6100 Darmstadt
Tel. 06151/82885

```
  1:  ;-------------------------------------------------
  2:  ;
  3:  ;                GEM-Demo 3
  4:  ;                Autor           : Marcus Kraft
  5:  ;                Letzte Änderung : 06.03.1990
  6:  ;
  7:  ;                (c) MAXON Computer GmbH 1990
  8:  ;-------------------------------------------------
  9:          .TEXT
 10:
 11:          .INCLUDE 'GEMLIB.IS'      ; GEM-Bibliothek
 12:                                    ; einbinden
 13:          appl_init                 ; Appl. anmelden
 14:
 15:          graf_handle               ; Bildschirm-
 16:                                    ; Handle holen
 17:          v_opnvwk   handle         ; Virtuellen Ar-
 18:                                    ; beitsplatz öff
 19:          graf_mouse #255,#maus_form ; Mausform 1
 20:
 21:          graf_mouse #256,#maus_form ; Hidemouse
 22:
 23:          vsf_color  handle,#1      ; Füllfarbe
 24:
 25:          vswr_mode  handle,#1      ; Zeichenmodus
 26:                                    ; Replace
 27:          vsf_interior handle,#2    ; Füllart Muster
 28:
 29:          vsf_perimeter handle#0    ; keine Umrah-
                                        ;   mung
 30:
 31:          vsf_style  handle,#9      ; Mauermuster
 32:
 33:          v_bar handle,fullscreen   ; ganzen Bild-
 34:                                    ; schirm füllen
 34:
 35:          move.w      #100,d6
 36:          move.w      #200,d5
 37:          moveq.l     #8,d7
 38:  loop:
 39:          vsf_style  handle,d7      ;Füllmuster
 40:
 41:          addi.w      #20,pxyarray
 42:          addi.w      #20,pxyarray+2
 43:          subi.w      #20,pxyarray+4
 44:          subi.w      #20,pxyarray+6
 45:
 46:          v_circle  handle,pxyarray,pxyarray+2,d6
 47:          v_circle  handle,pxyarray+4,pxyarray+2,d6
 48:          v_circle  handle,pxyarray,pxyarray+6,d6
 49:          v_circle  handle,pxyarray+4,pxyarray+6,d6
 50:          subi.w     #10,d6
 51:
 52:          v_circle   handle,#320,#200,d5
 53:          subi.w     #20,d5
 54:
 55:          subq.w      #2,d7
 56:          cmpi.w      #0,d7
 57:          bne         loop
 58:
 59:          vsf_style  handle,#9
 60:
 61:          v_fillarea handle,#13,polygon  ; Polygon
 62:                                         ; zeichnen
 63:          vsl_width  handle,#5     ; Linienbreite 5
 64:
 65:          vsl_ends   handle,#2,#2  ; abgerundete
 66:                                   ; Linienenden
 67:          v_pline handle,#13,polygon ; Polygonum-
 68:                                   ; rahmung z.nen
 68:
 69:          vsf_interior handle,#0   ; Füllart leer
 70:
 71:          vsf_perimeter handle,#1  ; Umrahmung
 72:                                   ; einschalten
 73:          v_rfbox handle,pxyarray  ; abgerundetes
 74:                                   ; Rechteck
 75:          vst_effects handle,#%10000 ; OUTLINE
 76:
 77:          vst_height handle,#40    ; Texthöhe 40
 78:
 79:          v_gtext handle,#200,#210,#demotext
 80:                                   ; Textausgabe
 81:          graf_mouse #257,#maus_form ; Showmouse
 82:
 83:          evnt_keybd               ; Auf eine
 84:                                   ; Taste warten
 85:          vsl_width  handle,#1
 86:
 87:          vst_height handle,#13
 88:
 89:          vst_effects handle,#0
 90:
 91:          v_clsvwk   handle        ; Virtuellen
 92:                          ; Arbeitsplatz schließen
 93:          appl_exit       ; Applikation abmelden
 94:
 95:          clr.w      -(sp)       ; Programm beenden
 96:          trap       #1
 97:
 98:
 99:          .DATA
100:
101:  demotext:     .DC.b "GEM für Alle!",0
102:                .EVEN
103:  fullscreen:   .DC.w 0,0,640,400
104:  pxyarray:     .DC.w 85,85,555,315
105:  polygon:      .DC.w 134,120,265,160,342,70,400,
                          150,525,125,477,210
106:                .DC.w 515,288,340,242,292,327,227,
                          241,111,272,160,200,134,120
107:
108:  maus_form:    .DC.w $0000,$0008,$0001,$0000,$0001
109:                .DC.w $0000,$1fff,$3fff,$7fff
                      ; Hintergrundmaske
110:                .DC.w $ffff,$f0f0,$f8f0,$7cf0
111:                .DC.w $3ef0,$1ff0,$0ff0,$fff0
112:                .DC.w $fef0,$fcf0,$f8f0,$0000
113:                .DC.w $0000,$0000,$1ffe,$3ffe
                      ; Vordergrundmaske
114:                .DC.w $7060,$6060,$7060,$3860
115:                .DC.w $1c60,$0e60,$0660,$0e60
116:                .DC.w $7c60,$7860,$0000,$0000
117:
118:                .END
```

Beispiellisting für die Verwendung von fastGEM

```
.MACRO v_gtext \1,\2,\3,\4
          movea.l      \4,a0         ; Adresse String
          movea.l      #intin,a1     ; Adresse INTIN
          clr.w        d1            ; Zähler löschen
\~loop:   move.b       (a0)+,d0      ; Byteweise verschieben
          move.w       d0,(a1)+      ; Wordweise verschieben
          addi.w       #1,d1         ; Zähler = Zähler + 1
          cmpi.b       #0,(a0)       ; Stringende erreicht ?
          bne          \~loop        ; Nein, dann weiter ...
          move.w       #8,contrl
          move.w       #1,contrl+2
          clr.w        contrl+4
          move.w       d1,contrl+6
          clr.w        contrl+8
          move.w       \1,contrl+12
          move.w       \2,ptsin      ; x-Koordinate
          move.w       \3,ptsin+2    ; y-Koordinate
          vdi
          .ENDM
```

Bild 4: Das Makro für die Textausgabe (VDI 8) läßt schon erahnen, wieviel Tipparbeit dem Benutzer durch fastGEM erspart wird.



Generationsfolge

P60 - neuer 24-Nadler von NEC

Wir erinnern uns. 1986 erscheint der NEC P6 und läutet für den Home-Bereich die Ära der 24-Nadler ein. Mit ihm setzt NEC einen Maßstab für Druckqualität. Zwei Jahre später folgt ihm der schnellere und komfortablere P6plus. Auf der CeBIT '90 zeigt uns NEC die nächste Generation des wohl erfolgreichsten 24-Nadlers überhaupt. Lesen Sie im Folgenden, ob der P60 den Vorsprung sichert.

Bei so ruhmreichen Ahnen ist das für den P60 keine leichte Aufgabe. Beginnen wir mit den Äußerlichkeiten. Rein konzeptionell betrachtet hat sich gegenüber dem P6plus wenig geändert. Der wuchtige P60 bietet in seinem geräumigen Innern bekannte Technik. Endloszufuhr von hinten, der große Schubtraktor befindet sich unter einer Klappe hinter der Walze. Das Einzelblatt wird über die hochzustellende Rutsche eingeführt. Währendessen harrt das endlose Druckgut in geparkter Position des nächsten Einsatzes. Die Bedienung erfolgt über Tasten, als Anzeige dienen LEDs und ein zweistelliges LED-Display.

En Detail

Wenn auch der NEC P60 nicht solche eklatanten Unterschiede gegenüber seinem Vorgänger besitzt wie dieser noch zum P6, fallen beim zweiten Hinsehen doch Feinheiten ins Auge oder besser: in die bedienende Hand. Die beiden Hebel an der Gehäuseoberseite (für Druckkopfabstand und Papiersorten) sind z.B. größer geworden.

Der dritte Hebel - beim P6plus noch zum Papiereinzug benötigt - ist durch die 'Load/Unload'-Taste an der Front ersetzt worden. Diese elegante Lösung wird dadurch ermöglicht, daß der Drucker keinen Andruckhebel mehr besitzt. Stattdessen befindet sich oberhalb des massiven Druckkopfes eine Reihe fester Rollen. Sie sehen diese auf dem Photo. Darunter manövriert die Mechanik das Papier beim Einzug. Die Rollen drücken das Blatt nicht nur an die Walze, sondern ziehen es auch ein wenig. Und beim Seitenvorschub schieben sie es ganz aus dem Druckwerk heraus.

Die Einzelblattrutsche rastet nun in der Arbeitsposition ein, beim Testgerät verhielt sie sich leider widerborstig. Die seitlichen Führungen auf ihr sind größer und geleiten das geschnittene Blatt sicherer ins Druckwerk als noch beim P6plus.

Das Gehäuse ist noch ein wenig größer, kantiger und im Farbton heller geworden. Wie Sie im Foto sehen, birgt es neben der soliden Mechanik auch viel Hohlraum. Die zweigeteilte Klappe an der Oberseite deckt die Öffnungen vollständig ab - durchaus mit Erfolg, denn die Geräuschentwicklung des NEC P60 ist recht erträglich geworden.

Befehle empfängt der Drucker nicht nur über die Schnittstelle, sondern auch per Folientaste. Diese sind noch immer von wenig anheimelnder Natur - das Display in gut lesbarem Grün ist ein schwacher Trost. Die FormFeed-Taste zum Vorschub der ganzen Seite fehlt. Dafür muß die Feed- Taste 2 Sekunden gehalten werden. Fünf Zeilenvorschübe brauchen demnach fünfmal Tippen...

DIP-Schalter gehörten bereits beim P6plus in den Bereich der Vergangenheit. Auch beim P60 werden die festen Einstellungen über ein Menü vorgenommen, wobei der Drucker auf dem Papier Frage und Antwort steht. Der Fortschritt besteht darin, daß dafür der Drucker nicht ausgeschaltet und die Einstellung nicht mehr explizit gespeichert werden muß. Trotzdem - bei einem Blick auf Drucker auch geringeren Preisniveaus - hätte ein alphanumerisches Display zur Konfiguration NECs jüngstem Kind besser zu Gesicht gestanden.

Au Travail

Bei der Arbeit fordert der P60 seinem Besitzer weniger Gewöhnung ab als seine Vorgänger. Das Einlegen von Endlospapier wie der Einzug der Einzelblätter - sie



Leicht geändert: Das Tastenfeld



Massiv: Das Innenleben



Grafikprobe: NEC P60 360 x 180 dpi

funktionieren sauber und ordentlich, wenngleich nicht ohne hier und da zu hakeln. Manchmal ist man versucht, an dem verlockend schön geformten Walzenknopf zu drehen. Schade, daß das verboten ist. Das Handbuch mahnt, wie bei jedem Drucker, zur ausschließlichen Benutzung der Feed- und Load- Tasten.

Trotz der Jahre, die ins Land gingen: Der P60 besitzt im Einzelblattbetrieb immer noch die Unart, häufig auf die Walze zu drucken. Nach dem Auswurf eines Blattes braucht nur das nächste den Papier-Sensor zu betätigen - der Drucker läßt sich OnLine schalten. Obwohl dann noch kein Papier vor dem Kopf ist, arbeitet die Maschine und färbt für 12 Zeilen die Hartgummiwalze.

Die Papierparkfunktion glänzt dagegen durch reibungslosen Betrieb. Sie bedarf keiner weiteren Kommentierung. Hilfreich ist die Funktion der Tear-Taste. Einige Geräte bieten solche Abreißhilfe nur als Automatik; hier kann man sie per Fingerdruck anfordern. Dabei wird das Papier vorgefahren, damit man es an der nächsten Perforation abreißen kann. Kurz darauf wird es soweit zurückgefahren, daß beim nächsten Druck kein oberer Rand bleibt. Das kann natürlich nur funktionieren, wenn man das Papier NIE zwischendurch abreißt, verlangt also Beherrschung. Ebenso hilfreich ist der Umstand, den Abstand der ersten Druckzeile vom oberen Papierrand einstellen zu können. Wählt man den Minimalwert, ist der P60 in der Lage, 67 (!) Zeilen auf ein DIN A4-Blatt zu drucken. Ein rekordverdächtiger Wert.

Schwarz und Weiß

Widmen wir uns jetzt den gedruckten Ergebnissen, die den neuen NEC verlassen. Geliefert wird er mit sieben verschiedenen Letter-Quality-Fonts. Damit steht dem Besitzer bereits ein reichlicher Vorrat an Schriften zur Verfügung. Darunter befinden sich allerdings drei reine Proportional-Fonts. Benötigt man in seinen Werken verschiedene Zeichenbreiten, bleiben nur noch vier Fonts zur Auswahl. Diese allerdings sind nicht in Proportionalschrift verfügbar... Ein Umstand, der bereits beim P6plus Anlaß zur Kritik gab. Darüber hinaus stehen Font-Karten zur Verfügung, die jeweils zwei Schriften enthalten. Vorn am Drucker eingesteckt, halten sie vor allem große Fonts und solche spezieller Art (technisch/ mathematisch/Barcode) bereit.

Sie sehen die verschiedenen Schriftformen in der Abbildung. Allen ist gemeinsam, daß sie sauber und gut lesbar sind. Es sind dies die gleichen Schriften wie bereits im P6plus und auch im P2plus. Dieses Detail ist für alle diejenigen interessant, die zum P60 auf- oder umsteigen wollen. Die äußere Form ihrer gedruckten Werke wird sich dann nicht ändern müssen.

Insgesamt läßt die Druckqualität des P60 wenig zu wünschen übrig. Das gilt vor allem für den Textdruck. NECs neuester erwies sich daneben als durchaus anschlagstarker Kollege. Zum Original auf selbstdurchschreibendem Papier druckt er noch locker vier lesbare Durchschläge. Betrachten Sie die Grafikprobe: Die senkrechten Striche sind leicht unscharf, die schwarze Fläche weist leichte helle Streifen auf. Ganz perfekt ist die Grafikqualität leider nicht.

Für den Druck in beiden Richtungen kann man im Menü des P60 den Versatz zwischen Druck von links und dem von rechts einstellen. Danach liefert er auch bidirektional gedruckte Strichgrafiken in akzeptabler Qualität. Darüber hinaus besitzt der P60 ein Feature, das besonders bei großen dunklen Flächen auffällt. Er ist nämlich in der Lage, sowohl dunkle Stellen in der Grafik zu erkennen als auch die Anschlagstärke der Nadeln zu steuern. Er vermindert diese dann und vermeidet so, daß sich das Papier - wie für Nadeldrucker typisch - wellt.

Hochgeschwindigkeit

Neben der Qualität des Ausdrucks fasziniert am NEC P60 vor allem seine Geschwindigkeit. Der Maschine beim Drukken zuzusehen, ist eine Freude. Werfen Sie einen Blick auf die Tabelle mit den Meßwerten. Deutlich erkennbar ist die Leistungssteigerung, die der P6-Familie im Laufe der Generationen widerfahren ist. Das betrifft sowohl den Text- als auch den Grafikdruck.

Die Texttests sind hinlänglich bekannt. Das lange Testdokument bringt der NEC in einer Geschwindigkeit zu Papier, die in bedrohliche Nähe zu Profi-Geräten rückt, die fast doppelt so teuer sind. Daß Signum! grafisch druckt, hat sich auch herumgesprochen. Trotz höchster Auflösung ist der P60 so flink, daß er den DIN-Brief in fast der gleichen Zeit druckt, wie der kleine Kollege aus gleichem Hause, der P2plus, das in normaler Briefqualität tut.

Die Grafiktests beweisen es: Momentan gibt es in dieser Preisklasse keinen Drukker, der Bilder so schnell druckt. Der Vorsprung vor dem P6plus sinkt zwar bei dem großen Testbild auf Null. Doch ist das durchaus kein Makel, denn auch dieser war bereits der eiligste Bilderproduzent seiner Klasse.

Wie heißt es doch: Dahinter steckt immer ein kluger Kopf. Tatsächlich bedarf es wohl einiger Maßnahmen, um einen Drucker so schnell zu züchten wie unseren Prüfling. Bei NEC ist man dabei über das plumpe Beschleunigen der Drukker- Hardware hinausgegangen. Wie die Spalte 'Kopfbeschleunigung' in der Tabelle bereits zeigt, optimiert die Maschine bei Texten mit Leerzeichen sehr stark. Zur Erklärung: Bei diesem Test werden zwei Druckzeiten verglichen. Zunächst die für 60 Zeilen mit je 80 Zeichen. Danach die Zeit für den Druck von 60 Zeilen mit je einem Zeichen vorn, 78 Leerzeichen und einem Zeichen am Zeilenende. Der Wert gibt den Geschwindigkeitszuwachs in

```
NEC P60
Hoch auf dem gelben Wagen sitz ich beim Schwager vorn.
Hoch auf dem gelben Wagen sitz ich beim Schwager vorn.
Hoch auf dem gelben Wagen sitz ich beim Schwager vorn.
```

Die Schriftprobe in Courier 10

```
Der Font ist ITC Souvenir 10
Der Font ist Helvetica 10 PT PS
Der Font ist Times 10 PT PS
Der Font ist New Prestige Elite 12WP
Der Font ist Bold PS
Der Font ist Quick Gothic 10
```

Fontvielfalt: Die eingebauten Fonts des P60

Prozent an. Er ist ein Maß für die Fähigkeit des Druckers, einerseits Leerstellen überhaupt zu erkennen, andererseits dafür, die Kopfgeschwindigkeit heraufzusetzen.

Dieses Feature erklärt sicher einen Teil der Geschwindigkeit des P60. Zum anderen ist sein Betriebssystem so aufgebaut, daß er gleichzeitig Daten aufnimmt und druckt. Das dürfte gerade an der langsamen Centronics-Schnittstelle des Atari ST zum Zuwachs führen. So steht das Druckwerk nicht, während eine Zeile übertragen wird, was bei Grafik ja schon mal über 4 kBytes (!) sein können. Die Gewichtung, welche der Aufgaben vorrangig ist, scheint er ebenfalls je nach Druckart zu treffen. Zumindest deutet die längere Zeit der Datenaufnahme im Draft-Betrieb darauf hin, daß er dem schnelleren Ausdruck dann die Priorität gibt. Nebenbei erwähnt, verfügt der P60 über einen Eingangsspeicher von 80 kBytes. Dieser macht bei vielen Aufgaben den Spooler überflüssig und erklärt die minimalen Zeiten zur Datenabnahme.

Alle Jahre wieder...

...finden verschiedene traditionelle Feste statt. Bei NEC braucht's zwei, bis ein neuer Drucker auf dem Markt ist. Immerhin eine kurze Entwicklungszeit. Als auf der CeBIT '86 die ersten P6 vorgeführt und kurz darauf geliefert wurden, betrug ihr Marktanteil im Herbst bereits 2,6%. Nach der Einführung des P6plus im Herbst '88 waren es bereits 4,9%. Mit Sicherheit wird der P60 auch dabei zulegen, bietet er doch wiederum ein Mehr an Leistung bei konstantem Preis (gegenüber den Vorgängern).

Doch die Konkurrenz hat in den letzten vier Jahren nicht nur zugeschaut, sondern bietet ihrerseits leistungsstarke 24-Nadler an, die mitunter vor allem komfortabler sind. Der Freak daheim schätzt wahrscheinlich vor allem die Leistung der Gerätschaften, mit denen er sich umgibt. Am Arbeitsplatz sieht das anders aus. Hier zählt die kurze Einarbeitungszeit, die Handlichkeit der Ausstattung. Denn die Motivation seiner Mitarbeiter kann man ganz einfach zerstören: indem man ihnen die falschen Werkzeuge gibt.

Geschwindigkeit ist also längst nicht alles. Bevor er begeistert zugreift, sollte der Kunde genau den zukünftigen Einsatz überdenken. Am NEC P60 überzeugen vor allem seine Geschwindigkeit und die robuste Verarbeitung; er birgt 10,5 kg solider japanischer Druckerbaukunst. Die Bedienung und Papierführung sind nach wie vor nicht über jede Kritik erhaben.

Bleibt schließlich nur zu erwähnen, daß für den Farbdruck eine Aufrüstung erhältlich ist. Für 570 DM mehr gibt es das breite Modell P70, das dann DIN A3-Format quer verarbeiten kann. Als Zubehör ist neben der automatischen Einzelblattzuführung auch ein Zugtraktor für schwieriges Druckgut (mehrlagige Durchschlagformulare etc.) erhältlich. NEC gibt auf seine neuen Modelle lobenswerterweise 12 Monate Garantie, die den Druckkopf mit einschließt.

IB

NEC P60 - unverb. Preise lt. Hersteller:

Drucker:	*DM*	*2154,60*
(P70: DM 2724,60)		
Einzelblatteinzug:	*DM*	*454,86*
(P70: DM 586,86)		
Zugtraktor:	*DM*	*313,50*
(P70: DM 340,86)		
Colorkit:	*DM*	*225,72*
serielle Schnittstelle:	*DM*	*225,72*
Font-Karten:	*DM*	*169,86*

Dokumentlänge Text: 33396 Bytes
Dokumentlänge Grafhik: 32643 Bytes
Genannt: Zeit zur Datenabnahme / Gesamtzeit für Druck
Datenmenge des Signum!-Briefes: 123000 Bytes (360 x 180 DPI)
Datenmenge des Schwarzweißbildes: 282240 Bytes (360 x 360 DPI)

Drucker	LQ-Einzel	LQ-Endlos	Draft-Endlos	Grafik	DIN 32751	Signum!-Brief	Sgnm!(360)	Bild 281KB	Kopfb.
NEC P6	-	8:20 / 10:49	4:30 / 5:50	0:30 / 0:31	0:40	1:10	2:14		0%
NEC P6plus	-	0:18 / 9:10	0:18 / 5:46 / 3:38 *	0:16 / 0:20	0:34	0:30	0:57	2:27	51%
NEC P60		0:197 / 7:48	0:25 / 4:25 / 3:34 *	0:17 / 0:18	0:30	0:26	0:47	2:27	89%
NEC P2plus	-	10:13 / 11:32	5:27 / 6:10 / 4:52 *	0:25 / 0:26	0:43	1:00	2:01	4:21	0%
OKI ML 390	-	5:20 / 7:53	2:20 / 3:27	0:24 / 0:25	0:28	1:43	3:25	3:44	0%

*: Die dritte Zeit ergibt sich bei Verwendung der High-Speed Draft-Qualität

Zahlen lügen nicht: Der neue P60 im Kreise seiner Vorfahren. Zum Vergleich der 'kleine Bruder' P2plus und der Konkurrent von OKI.



DELLCOM –

SCSI-Festplatten der Spitzenklasse

Leistungsmerkmale der Dellcom-Festplatten:

- Datentransferraten der verwendeten Festplatten bis zu 1,25 MB/sec. Mittlere Zugriffszeiten bis zu 14 ms, Interleave 1:1.
- 100% Atari-kompatibel, unterstützt sämtliche Fremdbetriebssysteme (PC-Speed, Supercharger, Spectre 128 GCR, Aladin, OS-9, RTOS), Festplatte autobootfähig; arbeitet sehr zuverlässig und superleise – für den professionellen Einsatz.
- Spitzensoftware: 12 Partitionen mit max. 512 MB pro Partition, Kennungen einstellbar für Atari, Macintosh oder IBM, einfache Bedienung. Ausführliches deutsches Handbuch. Integriertes Cache, Mediumwechsel-Erkennung. Harddisktreiber mit Verify-Funktion. Ordnererweiterung im Treiber integriert.
- Gehäuse aus Stahlblech gefertigt. Aussenmasse und Farbe wie Atari Mega ST's. Durchgeschleifter, gepufferter DMA-Bus. Alle Festplatten arbeiten problemlos mit dem Atari Laser. Hardwaremässige Autoparkfunktion. Herausgeführter SCSI-Bus (50poliger Centronics-Anschluss für Apple-Macintosh und IBM-PCs). Vorbereitet für zweite SCSI-Festplatte (im Gehäuse nachrüstbar). DMA-Adressen von aussen über Nummernschalter einstellbar. Eingebaute Uhr.
- Alle Dellcom-Festplatten werden komplett anschlussfertig mit Netz- und DMA-Kabel, deutscher Software, verschiedenen Hilfs- und Testprogrammen sowie deutschem Handbuch geliefert. Hotline- und Up-Date-Service. Garantie 1 Jahr. Dellcom-Festplatten werden in der Schweiz hergestellt.

Folgende Kapazitäten können geliefert werden:

Dellcom-Festplatten: 30, 40, 50, 100, 150, 209 MB;
Dellcom-Wechselplatten: 44, 175 MB.



Alpenstrasse 1, 3072 Ostermundigen, Tel. 031 31 00 32
Händler Anfragen erwünscht.

DER MAXON GAL-PROMMER



Das Entwicklungssystem für Logikschaltungen

Der **MAXON GAL-Prommer 16/20** ist das praktische und leistungsstarke Programmiergerät für die gängigen GAL-Typen 16V8 und 20V8. Möchten auch Sie diese modernen Bausteine nutzen, um Ihre Schaltungen schnell, preiswert und flexibel aufzubauen? Mit dem MGP 16/20 kein Problem, denn im ausführlichen Bedienungshandbuch wird auch erklärt, wie herkömmliche Logik (NOR-, NAND-, NOT-, ...-Gatter) in ein GAL programmiert wird und wie diese Bausteine arbeiten.

Neben der enormen Platz- und Stromersparnis bleibt eine einmal aufgebaute Schaltung flexibel, denn ein GAL ist in wenigen Sekunden umprogrammiert und kann somit neuen Anforderungen angepaßt werden. Einige typische Einsatzgebiete eines GALs sind Adreßdekoder, Zustandsautomaten, logische Gatter oder die **PAL-Simulation**.

Zusammen mit dem MGP wird in der neuesten Version ein vollständiger TwoPass Logic-Compiler ausgeliefert. Dieser Compiler übersetzt logische Gleichungen für die GALs 16V8 und 20V8 sowie deren A-Typen in die JEDEC-Dateien, die in das GAL programmiert werden. Dabei führt der Compiler optional eine Optimierung der Logikgleichungen nach Quine-McCluskey durch. Zusammen mit dieser Software-Erweiterung stellt der MGP ein komplettes Entwicklungssystem für logische Schaltungen am ATARI ST dar.

Easytizer - Der Videodigitizer ohne Geheimnisse

Mit dem **Easytizer** (Bestell-Nr.: 871000) können Sie beliebige Videosignale von einer Schwarzweiß- oder Farb-Kamera, Videorecorder oder direkt vom Fernsehgerät (mit Composite Video Ausgang) digitalisieren und somit auf dem Bildschirm Ihres ATARI ST sichtbar machen. Der Easytizer wird am Modul-Port des ST angeschlossen. Die Auflösung beträgt 800x600 Bildpunkte, so daß in mittlerer Auflösung 640x200 Bildpunkte in vier Graustufen dargestellt werden können. In dieser Betriebsart werden 12,5 Bilder in der Sekunde wiedergegeben. Im hochaufgelösten Modus werden 640x400 Bildpunkte in Schwarzweiß wiedergegeben. Einige Besondere Merkmale des Easytizers:

Software vollständig in Assembler • Von STAD und Sympatic Paint ansteuerbar • Ein beliebiger Bildausschnitt kann in ein mit der Maus wählbares Format vergrößert und verkleinert werden • Spiegeln eines Bildes in horizontaler und vertikaler Ebene • Animation mit beliebig vielen Bildern möglich, nur durch die Kapazität des Rechners begrenzt (beim Mega ST4 über 100 Bilder) • Eingebauter Druckertreiber für die mittlere Auflösung für NEC P6/P7 und EPSON oder Kompatible • Wahlweise automatische oder manuelle Helligkeitseinstellung • Schnappschuß

Erhältlich ist der Easytizer als Fertiggerät (komplett aufgebaut und geprüft, inklusive Diskette mit der Easytizer-Software und Bedienungsanleitung), oder als Teilesatz (Platine, programmiertes GAL 16V8, Quarzoszillatormodul 32 MHz, Diskette und Bedienungsanleitung).



MAXON Computer GmbH
Schwalbacher Straße 52 • 6236 Eschborn
Telefon 06196/481811 • FAX 06196/41885

DER MAXON JUNIOR PROMMER



Das universelle EPROM-Programmiergerät

Der **Junior Prommer** (Bestell-Nr.: 880310) programmiert alle gängigen EPROM-Typen (und deren CMOS-Typen), angefangen vom 2716 (2 kByte) bis zum modernen 27011 (1 MBit). Mit dem optionalen Mega-Modul-Adaptersockel (Bestell-Nr.: 880313) lassen sich sogar die neuesten 32-pol. EPROM-Typen 27010, 27020, 27040 und 27080 brennen. Aber nicht nur EPROMs, sondern auch einige ROM- und EEPROM-Typen lassen sich lesen bzw. programmieren.

Die komfortable Software bietet mannigfaltige Optionen, wie z.B. Zerlegung eines 16-Bit-Words in High- und Low-Byte, 5 Programmieralgorithmen für hohe Datensicherheit sowie einen funktionellenHex/ASCII-Monitor mit vielen Editier-Funktionen. Neben der Treibersoftware werden eine RAM-Disk, ein Programm zum Erstellen von EPROM-Karten, Lese- und Programmierroutinen sowie der Schaltplan mitgeliefert.



Übersicht der mit dem JuniorPrommer programmierbaren Speichertypen

Typ			
2716	25V	2732	25V
2732A	21V	2764	21V
2764A	12V	27128	21V
27128A	12V	27256	12V
27256	21V	27512	12V
27513	12V	27011	12V
✓ 27010	12V	27020	12V
27040	12V	27080	12V
4732		4764	
47128		47256	
X2804A		X2816A	
X2864A		X28256A	

DPE - Der universelle elektronische Drucker-Port-Expander

Der **DPE** ist eine vielseitige und funktionelle Erweiterungsschaltung für den Druckerport des ATARI ST. Welcher Besitzer von mehr als einem Zubehörgerät, das den Druckerport belegt, hat nicht schon einmal das dauernde Umstecken satt gehabt? Mit dem DPE ist damit Schluß! Der DPE ermöglicht es, elektronisch zwischen 3 verschiedenen Geräten umzuschalten. Für Geräte, die ihre Stromversorgung über den Joystickport beziehen, bietet der DPE zwei zusätzliche Joystickbuchsen. Da der DPE über eigene Treiberbausteine verfügt, können die angeschlossenen Geräte problemlos auch über größere Entfernungen angesteuert werden.

Erhältlich ist der DPE als Bausatz (doppelseitige Platine, Stückliste sowie Bauanleitung) unter der Bestell-Nummer 900800.

AUSFÜHRUNGEN UND PREISE:

• MGP-GAL-Programmiergerät	DM 195,00
• MGP-Platine, -Software und -Gehäuse	DM 95,00
• Junior Prommer ST (Fertiggerät wie beschrieben)	DM 229,00
• Adaptersockel Mega Modul f. Junior Prommer	DM 99,00
• Junior Prommer Leerplatine und Software (o. Bauteile)	DM 59,00
• Junior Prommer Leergehäuse (gebohrt und bedruckt)	DM 39,90
• ROM-Karte 128 kByte bietet maximal 4 EPROMs Platz (fertigbestückt o. EPROMs)	DM 58,00
• Easytizer (Fertiggerät)	DM 289,00
• Easytizer (Teilsatz wie oben beschrieben)	DM 129,00
• DPE Teilesatz	DM 59,00

Versandkosten:
Inland DM 7,50 — Auslandsbestellungen **nur** gegen Vorauskasse
Ausland DM10,00 — Nachnahme zuzgl. DM 4,00 Nachnahmegebühr.



ST SIMULA

Brandneuer Oldie

Simula ist eine objektorientierte Programmiersprache, die schon in den sechziger Jahren entstand, durch Standardisierung portable Programme ermöglicht und nun als ST-Simula den ATARI ST erreicht hat.

ST-Simula von Simula Team in Dortmund wird auf zwei Disketten mit einer Bedienungsanleitung und einer Sprachreferenz geliefert.

Installation

Für die Installation des Simula-Systems muß man etwas Zeit mitbringen. In einem ersten Schritt werden die Dateien der ersten Lieferdiskette in einen beliebigen Ordner kopiert. Die Dateien auf der zweiten Diskette sind mit dem bekannten Komprimierer ZOO gepackt und werden mit zwei Programmaufrufen automatisch auf die Platte kopiert. Dieser Vorgang dauert von Diskette auf Festplatte 45 Minuten - ein Glück, daß eine Installation nur einmalig nötig ist.

Man kann das System natürlich auch mit einem reinen Diskettensystem benutzen - hier werden insgesamt drei doppelseitige Disketten benötigt. Zum Auspacken sind ein zweites Laufwerk oder eine RAM-Disk nötig.

Nach der Installation befinden sich auf der Platte nahezu 225 Dateien in 16 Ordnern, die fast 2 Megabyte belegen. Die Hälfte der Dateien stellen allerdings Beispiele, die im normalen Betrieb natürlich nicht benötigt werden. Zum Betrieb von ST-Simula sollte man aber dennoch mit 2 MB Plattenplatz rechnen.

Oberfläche

ST-Simula ist ein "herkömmliches" kommandozeilenorientiertes Entwicklungssystem - im Gegensatz beispielsweise zu Turbo-C. Die Teilprogramme werden per Tastatur und unter Verzicht auf Mausbenutzung von der Guläm-Shell aufgerufen. Das Kernstück des Systems ist der Compiler SIMULA.TTP, der aus einem Simula-Programm einen Assembler-Quellcode erzeugt, und die Laufzeitbibliothek SIMULA.A mit Laufzeitsystem. Der "Rest" des Paketes ist aus vorhandenen Public Domain- und Shareware-Produkten übernommen.

Den Assembler-Code übersetzt der Assembler JAS aus dem SOZOBON-C-Paket in Objektmodule, die vom Linker LD - ebenfalls aus SOZOBON übernommen - mit dem Laufzeitsystem zu ausführbaren Programmen gebunden werden. Alternativ zu JAS und LD können auch MADMAC und ALN aus dem Atari-Entwicklungspaket übernommen werden. Als Shell kommt die Guläm-Shell zum Einsatz, die beispielsweise auch im Jefferson-Modula zu finden war.

Den Entwicklungsprozeß unterstützen zwei weitere Werkzeuge aus dem SOZOBON-Paket: MAKE für die automatische Übersetzung voneinander abhängiger Module und AR zur Erstellung und Verwaltung von Archiven, die Bibliotheksmodule schnell und platzsparend bereitstellen.

Die Übernahme von Produkten aus dem Freeware- und Public Domain-Bereich kann durchaus Sinn machen, immerhin handelt es sich bei den genannten Programmen um ausgetestete und zuverlässige Produkte. Damit vermeidet ST-Simula auch das Risiko von Bugs in diesen Systemkomponenten. Auf die Guläm und die SOZOBON-Komponenten soll hier nicht weiter eingegangen werden, wir hatten sie schon in PD-NEWS Nr. 3, respektive PD-NEWS Spezial vorgestellt.

Zur Edierung eines Simula-Programms kann ein beliebiger Editor zum Einsatz kommen, wobei daran zu erinnern ist, daß man aus der Guläm auch GEM-Programme einfach starten kann. Erheblich schneller ist natürlich die Verwendung des in Guläm eingebauten Micro-Emacs.

Die einzelnen Systemprogramme Compiler, Assembler und Linker müssen nicht "von Hand" nacheinander aufgerufen werden. Simula-Team liefert eine Reihe von Guläm-Scripts - also Batch-Dateien - mit, die die einzelnen Phasen der Programmerstellung automatisch ausführen. So enthält das Script simcl.g die Kommandos zum Aufrufen des Compilers, des Assemblers und des Linkers. Will man das Programm sofort ausführen, benutzt man das Skript simclg.g, wobei das "g" für "Go" steht. Damit werden die Entwicklungsphasen elegant und einfach erlernbar unterstützt.

Für größere Programme, die aus mehreren Modulen bestehen, wird man sich allerdings eher ein Makefile schreiben, das die Abhängigkeiten zwischen den Modulen beschreibt (Beispiele für Makefiles werden mitgeliefert). Der Aufruf von Make führt dann die notwendigen Compilier- und Linkvorgänge aus. Unter den Beispielprogrammen findet sich ein Programm namens SIMDESK, das eine Alternative zur Guläm-Benutzung andeuten soll. SIMDESK entspricht den von anderen Entwicklungssystemen bekannten mausgesteuerten Shells; so findet der Compiler-Aufruf hier mittels Menüauswahl statt.

Die kleine Shell ist allerdings noch im Stadium eines Prototypen - zwar voll funktionstüchtig, aber momentan eher eine Illustration der GEM-Programmie-

rung mit ST-Simula. Nach Angaben des Herstellers soll diese grafische Shell ausgebaut werden, ohne allerdings damit die Guläm abzulösen.

Compiler

Dem Compiler können verschiedene Kommandos als Optionen für den Übersetzungsvorgang mitgegeben werden. Mit ihnen kann man Zeitmeldungen über die Dauer der einzelnen Übersetzungsphasen erzeugen (Bild 1) oder Warnungen unterdrücken. Weiterhin lassen sich Übersetzungsprotokolle oder Fehler-Listings erzeugen. Für die abschließende Erzeugung eines fertigen Programms dient die Option -h, mit der keine Debugging-Informationen wie Vermerke über Zeilennummern im Quellcode mehr in den erzeugten Objekt-Code aufgenommen werden. Damit werden die Programme natürlich schneller und kürzer.

Zur Korrektur eines Programms aufgrund des Fehler-Listings wird man einen Editor mit mehreren Fenstern benötigen, also beispielsweise Tempus oder eben den Micro-Emacs aus der Guläm.

Ähnlich anderen Entwicklungssystemen wie Pascal oder Modula kennt auch Simula Direktiven, die in den Programmtext eingestreut werden. Die in ST-Simula implementierten Direktiven steuern die Paginierung und Erzeugung von Übersetzungsprotokollen sowie die Aufnahme von Debugging-Informationen in den Objekt-Code.

Mit "%INCLUDE <Datei>" läßt sich eine andere Simula-Datei in den Programmtext einlesen, womit man allerdings etwas in Konflikt gerät mit dem Modularisierungsgedanken. Zu beachten ist auch, daß die Direktiven noch nicht standardisiert und damit nicht portabel sind.

Etwas schwach zeigt sich die Fließkomma-Arithmetik. Ein kleiner Test mit der Quadrierung der Quadratwurzel von 2 ergibt nicht den ursprünglichen Wert, sondern 1.9... Turbo-C beispielsweise liefert das korrekte Ergebnis. So klein die Differenz sein mag, jeder Fehler setzt sich in komplizierteren Ausdrücken fort und wächst dabei. Eine Verbesserung der Fließkomma-Routinen hin zu mehr Genauigkeit ist eine dringliche Aufgabe für eine neue Version.

Die komplett gelinkten Programme haben eine erstaunliche Größe. Ein "leeres" Programm ohne jegliche Statements kommt immerhin auf cirka 52 KB. Ursache dafür sind nicht nur die Coroutinenunterstützung und der Garbage-Collector, die auf jeden Fall im Laufzeitsystem stehen müssen. Da die Standardklassen im-

```
Compiling...
simula -l -p hello
Lund Software STANDARD Simula Compiler Revision 4.07.2
ST-Version 2.1 by SIMULA Team Dortmund
Executing at ATARI ST 17-MAY-90 11:37:42
End of pass 1 0.240 seconds
End of pass 2 0.010 seconds
End of pass 3 5.620 seconds
End of pass 4 4.760 seconds
End of pass 5 0.000 seconds
End of pass 6 5.030 seconds

      No errors and no warnings.
      End of Simula Compilation.

Assembling...
jas hello.s

Linking...
ls -p hello.o d:\simula\simula.a
```

Bild 1: Die Bildschirmmeldungen während eines Compiler-Laufs.

```
Lund Software Simula Runtime System Revision 4.07.2
Pool size 64 K bytes.
...
Garbage Collection No. 1 starts. At source line 72
Released memory 55586 byte in 0.325 Seconds.
...
End of Simula execution. 61.605 CPU-Seconds, 0.325
used by 1 garbage collections
```

Bild 2: Die Meldungen über den Fortgang der Garbage-Collections

plizit als Präfixe für das Hauptprogramm stehen, ist beispielsweise das File-System immer in jedem Programm vorhanden, selbst wenn es nicht benutzt wird. Auch wenn bei größeren Anwendungen der Platzbedarf des Laufzeitsystems nicht entscheidend ist, wäre eine Optimierung auf Entfernung nicht benutzter Klassen durchaus wünschenswert.

Wir verzichten an dieser Stelle auf Benchmark-Tests. Da kein anderes Simula für den ST vorliegt, wären Simula-Benchmarks nur in Vergleich zu anderen Maschinen zu setzen - hier würde aber eher die Rechenleistung des ST gemessen als die Schnelligkeit des erzeugten Codes. Auch ein Vergleich mit Programmen aus anderen Sprachen würde unfair sein, da beispielsweise die Objektorientiertheit von Simula nicht als Pluspunkt für die Benchmarks verrechnet werden könnte. Nach Auskunft der Entwickler befindet sich die Code-Erzeugung in einer Überarbeitung.

Auch das Laufzeitsystem kann mit Optionen konfiguriert werden, die einfach beim Start eines fertig compilierten Programms als Kommandozeilen-Optionen mitgegeben werden. So würde der Programmaufruf "hello -p" von Guläm aus das Programm HELLO.TTP ausführen und dabei ein Protokoll über die Speicherverwendung auf dem Bildschirm ausgeben. Mit "-g" erhält man eine Mitteilung über die Durchführung von Garbage Collections, in denen der dynamisch verwaltete Speicher aufgeräumt wird. Eine Portierung des symbolischen, interaktiven Laufzeitdebuggers des Lund-Simula-Systems ist aber geplant.

Ein Beispiel für solche Protokolle sehen Sie in Bild 2. Da die Speicherverwaltung natürlich Einfluß auf die Laufzeit eines Programms hat, bieten diese Optionen eine gute Möglichkeit, das Laufzeitverhalten eines Programms zu überwachen und den Programmtext gegebenenfalls zu optimieren.

Bibliotheken

ST-Simula umfaßt Bibliotheken, respektive Klassen nach dem Simula-Standard. Hinzukommt ein kompletter Satz der ST-typischen Bibliotheksroutinen für AES, VDI, TOS und Line-A. Sie orientieren sich an den üblichen C-Bindings. Leider sind sie nur als Text-File auf der Diskette dokumentiert, man wird sich für die GEM-Programmierung also mit einem Stapel DIN-A4-Ausdrucken herumschlagen müssen. Hier wäre eine Dokumentation im Handbuch wünschenswert.

Auf die AES- und VDI-Bibliotheken soll übrigens ein objektorientierter Aufsatz gesetzt werden, mit dem dann die GEM-Programmierung einfacher werden soll. Diese Schnittstelle liegt momentan allerdings noch nicht komplett vor und ist nicht dokumentiert.

Dokumentation

Als Dokumentation werden eine Bedienungsanleitung und eine Sprachreferenz im DIN-A5-Format geliefert. Die Bedie-

Simula - alt, aber hochmodern

Simula wurde schon Anfang der sechziger Jahre, ausgehend von ALGOL, entwickelt. Der heute noch gültige Standard entstand 1967; man spricht inzwischen aber nicht mehr von Simula 67, sondern läßt die Jahreszahl einfach weg.

Als Urheber gelten die beiden Norweger Ole-Johan Dahl und Kristen Nygaard. Ihre Arbeit am Norwegian Computing Center in Oslo führte schließlich zu einem Standard, der eine hohe Kompatibilität zwischen verschiedensten Simula-Implementierungen sichert. Obwohl damit schon über zwanzig Jahre alt, enthält Simula verschiedene Konzepte, die heute als hochmodern gelten. Wir wollen ein paar davon herausgreifen.

Simula ist objektorientiert und das schon zu einer Zeit, als sich dieser Begriff noch nicht verbreitet hatte. All das, was heute beispielsweise mit C++ als Offenbarung gefeiert wird, ist in Simula schon vor langen Jahren implementiert worden.

Dazu einige Beispiele aus [1]. Eine Klasse besteht aus einer Menge von Attributen. Dies können Prozeduren oder Variablen sein, wodurch eine Kapselung von Daten und dazugehörigen Prozeduren strukturiert möglich ist. Ein Klassenobjekt wird zur Laufzeit erzeugt, wobei all diese Attribute neu entstehen. Von außen können die Attribute durch eine Referenz angesprochen werden. In unserem Beispiel soll eine Klasse exchange definiert werden, die bei der Erzeugung zwei Parameter erhält und diese danach vertauscht als Attribute anbietet:

```
class exchange(a,b); real a,b;
begin real c;
  c:=a;
  a:=b;
  b:=c;
end;
ref (exchange) swap;
real d,e;
  ...
  swap:-new exchange(4.5,3.2);
  d:=swap.a;
  e:=swap.b;
  ...
```

exchange ist nun eine Klasse, die die Attribute a und b nach außen bereitstellt. swap wird als eine Referenz deklariert, mit der im Programm ein Klassenobjekt vom Typ exchange erzeugt wird, dessen Attribute angesprochen werden können. Mit dem new wird ein Klassenobjekt vom Typ exchange erzeugt. Dabei werden zwei Parameter übergeben, auf die sofort bei der Erzeugung das Programm der Klasse ausgeführt wird. Die drei Statements vertauschen einfach die zwei übergebenen Werte.

Bei der Erzeugung entsteht eine Referenz auf die neue Klasseninstanz, die in swap abgelegt wird. Man kann sich eine Referenz vielleicht ähnlich einem Zeiger in anderen Programmiersprachen vorstellen, nur daß eine Referenz nicht etwa eine Speicherstelle referiert, sondern alle Attribute einer Klasse.

Die zwei Zuweisungen an d und e greifen nun mit der Referenz swap auf die zwei Klassenattribute a und b zu. Da bei der Erzeugung der Klassenobjekte die zwei Werte vertauscht werden, steht in d dann 3.2 und in e der Wert 4.5. Das Beispiel mag zwar nicht sehr sinnvoll sein und nicht unbedingt Klassen erfordern, zeigt aber doch die Mechanismen des Klassenkonzepts von Simula.

Hat man einige Klassen, die sich vielleicht logisch ähneln und damit teilweise dieselben Attribute enthalten, kann man die gemeinsamen Attribute in einer Klasse zusammenfassen und sie durch Prefixing in die anderen übernehmen. Ein weiteres Beispiel aus [1]:

```
class vehicle;
begin real length, width, speed;
end;

vehicle class bus;
begin integer seats, passengers;
end;

vehicle class truck;
begin real load, capacity; end;
```

Hier enthält die Klasse vehicle Attribute, die jede Art von Fahrzeugen hat, nämlich Länge, Breite und Geschwindigkeit. Die Klassen bus und truck sind Fahrzeuge, übernehmen durch Prefixing die Attribute von vehicle und fügen eigene Objekte hinzu. So sind in der Klasse bus zusätzlich die Attribute Sitze und Passagiere vorhanden.

Wir können das Beispiel so erweitern, daß Listen von verschiedenen Fahrzeugen verwaltet werden können. Für Listen ist standardmäßig die Kontextklasse Simset mit ihren Klassen head und link zuständig. Wir prefixen vehicle einfach mit link und können nun alle Listen-Operationen mit allen Fahrzeugen durchführen:

```
link class vehicle;
...
ref (head) list;
list :- new head;
new bus.into(list);
new truck.into(list);
new bus.int(list);
```

list wird als Referenz vom Typ head deklariert. head ist praktisch der Anker einer Liste. Nach der Erzeugung einer Liste mit new können die verschiedenen Fahrzeuge mit into in die Liste eingefügt werden. Da die Klasse link Präfix von vehicle war, gibt es in der Klasse bus und in truck das Objekt into.

Die zweite Besonderheit an der Sprache Simula ist ihre namensgebende Eignung für Simulationen. Dazu stellt die Standardklasse Simulation die nötigen Routinen bereit.

Die Aktivitäten der Simulation werden als Prozesse dargestellt. Diese können zu bestimmten Zeiten aktiviert und ausgeführt werden. Dabei werden sie entlang der simulierten Zeitachse entsprechend dem Simulationsmodell aktiviert.

Für die Aktivierung eines Prozesses sorgen hauptsächlich die activate-Statements. activate p at 100.0 aktiviert den Prozess zum Zeitpunkt 100 auf der simulierten Zeitachse. Mit activate p delay 100.0 würde p 100 Zeiteinheiten nach der aktuellen simulierten Zeit aktiviert werden.

Aufgrund dieser Statements ist bei der Ausführung eine Schedule-Liste entstanden, aufgrund der die einzelnen Prozesse zur Ausführung kommen. Damit beherrscht Simula ein Coroutinen-Konzept, das von der Simulation-Klasse zu einem Simulations-Modell mit einer Zeitachse erweitert wird. Diese Beschreibung ist natürlich verkürzend, und für ernsthafte Anwendungen ist eine gewisse Kenntnis von Simulationsmodellen nötig.

Engagierte Anwender haben sich in einer Simula-Benutzergruppe zusammengeschlossen, die vierteljährlich einen SIMULA-Newsletter herausgibt:

Association of SIMULA Users (ASU)
Ron Kerr
Computing Laboratory
University of Newcastle upon Tyne
GB-Newcastle upon Type NE1-7RU
Großbritannien

E-mail: r.kerr@newcastle.ac.uk

[1] P.R.Hills: An Introduction to Simulation Using Simula. Norwegian Computing Center, Oslo, 1973.

nungsanleitung beschreibt den Installationsvorgang, die Benutzung des Systems und den Compiler, den Linker und das Laufzeitsystem sowie die Implementationsdaten und -restriktionen. Die Einbindung in Guläm und die Shell-Skript werden mit Beispielen erläutert. Übrigens werden die Dokumentationen der Public Domain- und Shareware-Produkte auf Diskette mitgeliefert.

Die Sprachreferenz ist eine Übersetzung des schwedischen Standards über Simula. Als solcher ist er sicherlich schwerer lesbar als ein Lehrbuch, gibt allerdings zwangsläufig Auskunft über alle Fragen zu Simula. Erfreulicherweise ist der Text auch ab und zu mit Beispielen versehen. Wie oben schon genannt, sind die ST-spezifischen Bibliotheken nicht auf Papier dokumentiert.

Eines wird die Dokumentation sicherlich nicht leisten können: Simula lehren. Wer also ohne Vorkenntnisse mit dem System arbeiten will, kommt um die Anschaffung eines Lehrbuches - von denen der Markt eine Menge bietet - nicht herum. Ebenso muß der Desktop-Benutzer zunächst den Umgang mit Guläm lernen. An diesem Punkt sind Englischkenntnisse nötig, da die Guläm-Anleitung nicht übersetzt wurde.

Preise

ST-Simula kostet den "Normalverbraucher" DM 198,- zuzüglich Porto und Verpackung. Für den Ausbildungsbereich gibt es einen ermäßigten Preis von DM 148,-, der beispielsweise für Schüler und Studenten, aber auch für Institutionen wie Schulen und Universitäten gilt.

Ein festes Update-Verfahren besteht noch nicht, allerdings soll bei kleineren Fehlerkorrekturen der Anwender nur die Selbstkosten tragen müssen. Für erweiterte neue Versionen wird allerdings eine höhere Gebühr entstehen. Eine spezielle Version zur Benutzung eines mathematischen Coprozessors ist geplant; sie soll allerdings preislich höher liegen.

Fazit

Mit ST-Simula ist eine langerprobte und standardisierte objektorientierte Sprache für den ST erschienen. Durch die übermäßige Programmgröße ist sie für die Programmierung kleiner Utilities eher nicht geeignet, und auch die Fließkomma-Arithmetik bleibt zu verbessern.

Wer aber komfortabel ernsthafte Anwendungen, insbesondere aus dem Bereich der Simulation, schreiben will, ist bestens bedient. Durch die Standardisierung lassen sich auch auf einem ST Programme schreiben, die auf andere Systeme portabel sind. Damit wird das System insbesondere für den Universitätsbereich interessant. Schließlich macht es der überaus günstige Preis zu einer wirklichen Alternative zu anderen Entwicklungspaketen.

RT

Bezugsadresse:

SIMULA-Team GmbH iG
Postfach 50 01 63
D-4600 Dortmund 50



Schwieriger als andere Programmiersprachen ?

Diese Frage liegt auf der Hand, wenn von einer für den Leser neuen Programmiersprache die Rede ist, und deshalb soll sie auch gleich beantwortet werden: Nein! Schlicht und einfach nein. Jedenfalls solange man sich auf die Dinge beschränkt, die man bisher schon in einer anderen strukturierten Sprache gemacht hat, sei es nun Pascal, Modula, C oder eine andere Sprache. Schwieriger wird es erst, wenn man auch die Spezialitäten von SIMULA ausnutzen möchte. Aber das hat ja noch etwas Zeit.

SIMULA hat sich einen Namen gemacht auf dem Gebiet der Simulation - wie sollte es auch anders sein -, und nach und nach spricht sich herum, daß SIMULA als erste Objektorientierte Programmiersprache auch hier immer noch führend ist. Weniger bekannt ist jedoch, daß SIMULA die komplette Sprachdefinition von Algol (bis auf wenige exotische Details) einschließt, und man somit in SIMULA auch ganz einfach wie gewohnt konventionell programmieren kann. Und das sogar noch wesentlich bequemer als in einer der später von Algol abgeleiteten Sprachen wie z.B. Pascal. (Zu diesem Thema sagte einmal der bekannte Informatik-Pionier C.A.R. Hoare spöttisch: Die Sprache Algol war eine deutliche Verbesserung gegenüber den meisten ihrer Nachfolger.)

Sie können sich das nicht vorstellen? Bitte sehr: SIMULA bietet - neben Simulation und OOP - im Bereich der konventionellen Programmierung u.a.

- eine echte Blockstruktur, so daß in einem inneren Block Arrays (Felder) mit der gerade benötigten Größe angelegt werden können,
- ein sinnvoll 'aufgeweichtes' Typkonzept, bei dem z.B. Integer und Real kompatibel sind (bei der Zuweisung von Real nach Integer wird automatisch gerundet) und bei Arrays als Parameter die Grenzen nicht festgelegt werden müssen,
- statt unflexibler Strings den Typ Text, der die Bearbeitung von Zeichenketten wesentlich erleichtert,
- weitere Raffinessen wie bedingte Ausdrücke und zusätzliche Parameterübergabemechanismen wie 'call by name',
- eine voll im Standard definierte Dateiverarbeitung, die auch Direktzugriffsdateien einschließt, so daß selbst Datenbanken portabel implementiert werden können,
- einen als Standardmodul definierten Listenmechanismus für die komfortable Verwaltung von LIFO-, FIFO- und anderen Strukturen,
- eine automatische Speicherverwaltung mit Hilfe eines Garbage-Collectors, der von selbst dafür sorgt, daß nicht mehr benötigte Daten entfernt werden, wenn der Arbeitsspeicher knapp wird,
- und last not least eine getrennte Übersetzung, die diesen Namen auch verdient, also auch mit voller Schnittstellenüberprüfung durch den Compiler.

Allein schon mit diesen Eigenschaften stellt sich SIMULA als attraktive Alternative zu weniger großzügig ausgestatteten Compilersprachen dar. Dazu kommen noch die ausgezeichnete Fehleranalyse durch Compiler und Laufzeitsystem (das Laufzeitsystem gibt so genaue Fehlermeldungen, daß sich in der Praxis die Frage nach einem Debugger gar nicht erst stellt) sowie die gute Portierbarkeit - beispielsweise zwischen Atari ST, PC, diversen UNIX-Systemen und Mainframes - und die professionelle Qualität aller SIMULA-Implementierungen, eingeschlossen der für den ST.

Einige Features wird allerdings der routinierte Pascal- oder Modula-Programmierer erst einmal vermissen: Das sind die Sets sowie das Dreigestirn Aufzähltyp, varianter Record und Case-Anweisung. Natürlich lassen sich diese Dinge mit anderen Konstrukten nachbilden, aber viel sinnvoller ist es, in diesen Fällen von der OOP einige Anleihen zu nehmen: Denn da gibt es wesentlich elegantere Lösungen. Doch mehr dazu in einer der nächsten Ausgaben, denn vorläufig wollen wir uns erst noch auf bekanntem Gebiet bewegen.

Damit Sie einen Eindruck davon bekommen, wie ein typisches nicht-objektorientiertes SIMULA-Programm aussieht, haben wir ein kleines Beispiel abgedruckt. Es handelt sich dabei um ein Programm, das eine (quasi) beliebige Anzahl von Zeilen einliest und diese dann mit Hilfe von Quicksort sortiert wieder ausgibt.

Auf den ersten Blick dürfte den meisten Lesern sicherlich eine gewisse Ähnlichkeit zu Pascal auffallen. Dies ist auch kein Wunder, wenn man weiß, daß beide Sprachen auf Algol zurückgehen. Sehen

wir uns zuerst einmal das Hauptprogramm SORT.SIM etwas näher an. Es ist in Klein- und Großbuchstaben geschrieben. Diese Schreibweise dient nur der besseren Lesbarkeit, sie hat aber genauso wie bei Pascal keine Bedeutung für den Compiler.

Ein SIMULA-Programm fängt mit BEGIN an und hört mit END auf, danach kommt nichts mehr, auch kein Punkt. Bei unserem Beispiel stehen vor dem eigentlichen Programm noch Deklarationen für externe Prozeduren. Diese könnten auch erst nach dem ersten oder in diesem Fall sogar erst nach dem zweiten begin (Zeile 15) deklariert werden. Durch die External-Deklaration werden externe Prozeduren (oder Klassen) dem Compiler bekannt gemacht. Die Schnittstellen werden dabei nicht angegeben, da sich der Compiler diese Informationen selbst aus den Attribut-Files beschafft, die er bei der Übersetzung der externen Module anlegt. (Dieser Mechanismus ist ebenfalls durch den Standard festgelegt, funktioniert also auf jedem Rechner gleich.) Daraus folgt, daß die Module eines Programms immer in Bottom-Up-Reihenfolge übersetzt werden müssen, da sonst Schnittstellenänderungen unberücksichtigt blieben. Zum Glück gibt es keine Chance, gegen diese Regel zu verstoßen, ohne daß dies der Compiler oder spätestens der Linker merkt.

Das Hauptprogramm besteht im Prinzip aus einem Block. Ein Block, was ist denn das schon wieder? Nun, ein Block ist nichts anderes als ein Programmstück mit einem eigenen Gültigkeitsbereich für Deklarationen, ähnlich einer Prozedur, nur daß ein Block überall da verwendet werden kann, wo auch eine Anweisung erlaubt ist. Ein Block hat aber im Gegensatz zu einer Prozedur keinen Namen und keine Parameter. Er besteht aus einem Deklarations- und einem Anweisungsteil, wobei der Anweisungsteil implizit mit der ersten Anweisung des Blocks beginnt, also syntaktisch nicht besonders gekennzeichnet ist. Nach der ersten Anweisung darf keine Deklaration mehr folgen, das wäre ein Fehler.

Wozu Blöcke zu gebrauchen sind, sehen wir gleich: Zum sortieren wird ein Array (Feld) in noch unbekannter Größe benötigt. Diese Größe wird mit Hilfe der Funktionsprozedur IntInput von der Tastatur eingelesen. Da nach dieser Anweisung keine Deklarationen mehr erlaubt sind, wird zum Anlegen des Arrays ein innerer Block benötigt, der in Zeile 15 beginnt. In den Arraygrenzen dürfen alle Größen verwendet werden, die konstant oder außerhalb des betreffenden Blocks deklariert sind, also auch die gerade erst belegte Variable max.

```
— INTINPUT.SIM ——————————————————
   1      2    integer procedure
   3    IntInput (Question);
   4  ! ———          ;
   5      text    Question ;
   6    begin
   7      OutText(Question); BreakOutimage;
   8      InImage;
   9      IntInput:= InInt;
  10    end IntInput;


— TEXTINPU.SIM ——————————————————
   1      2    text procedure
   3    TextInput (Question);
   4  ! ———          ;
   5      text     Question ;
   6    begin
   7      OutText(Question); BreakOutimage;
   8      InImage;
   9      if not (SysIn.EndFile or
  10              SysIn.Image.Sub(1,3)="///")
  11            ! da sysin.endfile bei Atari-ST immer false;
  12      then
  13        TextInput:- Copy(SysIn.Image.Strip);
  14    end TextInput;


— INTTEXT.SIM ——————————————————
   1      2    text procedure
   3    IntText  (i, w);
   4  ! ———          ;
   5      integer i, w ;
   6    begin
   7      text t;
   8      t:- Blanks(if w <= 0 then 10 else w);
   9      t.PutInt(i);
  10      IntText:- t;
  11    end IntText;


— QUICKSOR.SIM ——————————————————
   1      2    procedure
   3    QuickSort    (a,l,r);
   4  ! ———          ;
   5      text array  a      ;
   6      integer        l,r ;
   7    begin
   8      text        x,t;
   9      integer     i,j;
  10      x:- a(l+(r-l)//2);
  11      i:= l; j:= r;
  12      while i<=j do
  13      begin
  14        while a(i)<x do  i:= i+1;
  15        while a(j)>x do  j:= j-1;
  16        if i<=j then
  17        begin
  18          t:-     a(i);
  19          a(i):-  a(j);
  20          a(j):-  t;
  21          i:=     i+1;
  22          j:=     j-1;
  23        end *** vertauschen *** ;
  24      end *** durchlauf *** ;
  25      if l<j then quicksort(a,l,j);
  26      if i<r then quicksort(a,i,r);
  27    end *** quicksort *** ;


— SORT.SIM ——————————————————
   1
   2    external text procedure      IntText,
   3                                  TextInput;
   4
   5    external integer procedure    IntInput;
   6
   7    external procedure            QuickSort;
   8
   9  BEGIN
  10
  11    integer max;
  12    while max<2 do
  13      max:= IntInput("Maximale Elementanzahl: ");
  14
```

→

```
15     begin ! *** innerer block *** ;
16
17       text array  elem(1:max);
18       boolean     fertig;
19       integer     i, last;
20       text        t;
21
22       while not fertig do
23       begin
24         i:= i+1;
25         t:- TextInput("Element" & IntText(i,5) & ": ");
26         if t =/= NoText then
27            elem(i):- (t & UpCase(Copy(t)))
                        .Sub(t.Length+1,t.Length)
28         else              begin  fertig:= true; last:= i-1  end;
29         if i=max then     begin  fertig:= true; last:= max  end;
30       end  *** einlesen *** ;
31
32       QuickSort(elem,1,last);
33
34       OutText("Sortierte Elemente:"); OutImage; OutImage;
35       for i:= 1 step 1 until last do
36       begin
37         OutText(elem(i).Main.Sub(1,elem(i).Length)); OutImage;
38       end  *** ausgeben *** ;
39
40     end    *** innerer block *** ;
41
42   END
```

Der Typ der Arrayelemente wird dem Schlüsselwort array vorangestellt. In SIMULA gibt es die Standardtypen boolean, integer, short integer, real, long real, character und text. Daneben gibt es die Objekttypen, die bei der Objektorientierten Programmierung die zentrale Rolle spielen, und die die im Vergleich zu Pascal fehlenden Typen mehr als ersetzen.

Als erstes werden vom Programm die zu sortierenden Zeilen in einer Schleife mit Hilfe der Funktionsprozedur TextInput eingelesen, und zwar entweder so viele wie hineinpassen, oder bis der Benutzer ein Endekennzeichen eingibt. Neben den allgemein üblichen While-Schleifen gibt es in SIMULA noch die For-Schleifen, die z.B. folgende Form haben können:

```
for i:= 1 step 1 until max do  ... ;
for i:= max step -1 until 1 do ... ;
```

oder auch:

```
for i:= 2, i+2 while i<max, i-2 while i>1 do
... ;
```

Hier können also die raffiniertesten Durchläufe mit Hilfe einer Anweisung programmiert werden. Dafür gibt es allerdings keine Repeat-Until-Schleife, aber die kann ebenfalls mit einer For-Schleife nachgebildet werden:

```
for i:= 1, 1 while ... do  ... ;
```

An dieser Stelle läßt es sich nun nicht mehr länger aufschieben: Wir müssen endlich ein paar Worte über den Typ Text verlieren. Das ist nämlich ein ganz besonderes Ding - nicht einfach nur eine Zeichenkette. Das ist nur der eine Teil von der Geschichte, und auch nicht ganz: Zu der Zeichenkette gehört noch die Längenangabe, die beide zusammen das Textobjekt bilden. Der andere Teil von der Geschichte ist die Textvariable. Eine Textvariable bezeichnet ein ganzes Textobjekt oder auch nur einen beliebigen Ausschnitt davon. Dies kann beispielsweise durch die Anweisung

```
t2:- t1.Sub(Startpositon,Länge);
```

geschehen. Die Textvariable enthält auch die Länge dieses Ausschnitts und eine aktuelle Bearbeitungsposition, so daß z. B. mit c:= textvariable.GetChar immer das nächste Zeichen gelesen werden kann, wobei durch GetChar die Position automatisch weitergeschaltet wird. Texte können auch wieder mit '&' verkettet werden.

Dies wird bei dem Aufrauf von TextInput (25) benutzt, das als Parameter einen Text hat, der aus drei einzelnen Texten zusammengesetzt wird. Dabei wandelt IntText wie der Name schon sagt einen Integer-Wert in einen Text um. Werfen wir bei dieser Gelegenheit gleich einen Blick auf IntText. Die Prozedur erhält als Parameter die Zahl i und die Anzahl der auszugebenden Stellen w. Zuerst wird mit Blanks ein leerer Text mit der benötigten Länge erzeugt. Dies ist entweder die angegebene Stellenzahl w oder - falls w zu klein ist - ersatzweise 10. An dieser Stelle kann man übrigens einen bedingten Ausdruck bewundern, durch den sich manches eleganter formulieren läßt. Zum Umwandeln eines Integer-Werts in eine Zeichenkette wird die Textfunktion PutInt (Zeile 9) verwendet. Entsprechende Funktionen gibt es auch für Real und für die umgekehrte Richtung. Dadurch kann man also auch eine Reihe von Zahlen erst einmal als Text einlesen und erst später irgendwo anders im Programm bei Bedarf in den gewünschten numerischen Typ konvertieren.

Etwas schwieriger zu verstehen ist da schon der Bandwurm von Zeile 27 des Hauptprogramms. (Keine Angst, der ist nur so kompliziert geraten, um einige Raffinessen zeigen zu können. Es geht selbstverständlich auch einfacher.)

Vorausgeschickt werden sollte erst einmal die Idee, die dahintersteht. Da die Sortierung unabhängig von der Groß- und Kleinschreibung sein sollte, ist eine Konvertierung in Großbuchstaben notwendig. Dies kann mit der Standardprozedur UpCase geschehen. Aber ausgeben wollen wir die Zeilen so, wie sie eingelesen wurden. Das bedeutet, wir müssen die Originalzeilen ebenfalls speichern und einen Bezug darauf von den sortierten Zeilen her festhalten. Dies ließe sich mit einem Record (bzw. in SIMULA mit einer Klasse) machen, der beide Texte enthält. Um den Umgang mit Teiltexten zu zeigen, haben wir uns für eine andere Lösung entschieden: Wir speichern in einem Text die Original- und die konvertierte Zeile, sortieren den konvertierten Teiltext, und geben am Ende den originalen Teiltext aus. Zuerst wird mit

```
t & UpCase(Copy(t))
```

der Gesamttext gebildet, der im linken Teil die Zeile unverändert, und im rechten Teil in Großbuchstaben enthält. Das Copy ist notwendig, weil UpCase auch seinen Parameter umwandelt und sonst beide Hälften nur Großbuchstaben enthälten würden. Mit

```
(...).Sub(t.Length+1,t.Length)
```

wird daraus der rechte Teiltext herausgeschnitten. Die Zuweisung an Textvariable erfolgt schließlich mit ':-', denn ':=' ist für Textwert-Zuweisungen vorgesehen. Nach dem Sortieren wird dann der linke Teiltext ausgewählt (Zeile 37). Dazu ist noch ein Zwischenschritte notwendig, es muß nämlich erst mit Hilfe von Main aus dem rechten Teiltext der Gesamttext gewonnen werden, aus dem dann analog der linke Teiltext herausgeschnitten wird.

Der Modul BasicIO steht in jeder Übersetzungseinheit automatisch zur Verfügung und stellt u.a. die Dateien SysIn und SysOut, die normalerweise mit Tastatur bzw. Bildschirm verbunden sind, schon geöffnet zur Verfügung. Dies geschieht



Hardware	
Atari 1040 STFM + SM 124	1 199,-
Atari 1040 STE + SM 124	1 498,-
Atari Mega ST 1 + SM 124	1 498,-
Atari Mega ST 2 + SM 124	2 198,-
Atari Mega ST 4 + SM 124	3 198,-
Atari Megafile 30	879,-
Atari Megafile 60	1 298,-
Epson-Drucker	
LX-400	429,-
EBZ LX-800 LQ 400/500	198,-
LQ-400, 24 Nadel DIN-A4	679,-
LQ-550, 24 Nadel DIN-A4	798,-
NEC P6+, dito	1 098,-
NEC P7+, dito DIN-A3	1 479,-
Farb-Option P6+ / P7+	279,-
Star LC-10	429,-
Star LC 24-10	679,-
EBZ LC 10 / 24	229,-
Software Atari ST	
GFA-BASIC 3.0 EWS ST	179,-
GFA-BASIC 2.0 EWS ST	44,-
GFA-Assembler ST	135,-
GFA-Draft plus ST	309,-
NEU! Turbo C 2.0 ST	198,-
NEU! Turbo C 2.0 Pro ST (T-C 2.0 / T-Debugger 1.0 & MAS-68K 1.2)	389,-
PC-Speed	439,-
Signum!Zwei	369,-
Stad	159,-
Megamax C	349,-
Modula 2	349,-
Superbase Professional	359,-
Superbase 2	179,-
LDW Power-Calc	219,-
Publishing Partner d	219,-
Cyber Paint 2.0	109,-
Devpac Assembler 2.0	128,-
CADproject Professional d	539,-
Schutzhauben aus hochwertigem Kunstleder Anthrazit	
ATARI 1040 / 260 / 520	24,95
Floppy SF 314 / 354	22,95
Monitor SM 124 / 125	27,95
Mega ST-Tastatur	24,95
Mega ST-Set	49,95
Epson-Drucker	
FX86/800/LX86/LQ500/850	24,95
FX1000/1050/LQ1050/2550	27,95
NEC-Drucker P6/7 P6/7 plus	27,95
Star-Drucker NL10/LC10/10c/24-10	24,95
Panasonic-Drucker 1080/91/1092/1592	24,95

Versand nur per NN, zzgl. Versandkosten
Abholung nur nach tel. Voranmeldung

TORNADO Computer Vertrieb
Wangenerstraße 99, 7980 Ravensburg
Tel. 0751/3951 Fax 0751/3953



DTP ohne Grafik – kaum denkbar.

Aber welche Grafik-Software kann schon große, hochauflösende Rastergrafiken verarbeiten? Welche unterstützt Vektorgrafik? Eigentlich müßte sie sogar beides können. Natürlich muß sie mit einem Großbildschirm arbeiten und darf nicht mehr als 300 Mark kosten.

Können Sie mit **Arabesque** alles haben. Rastergrafiken beliebiger Größe, Vektorgrafiken mit 1440 dpi Auflösung und die Möglichkeit, beides zu kombinieren. In einem Programm.

Arabesque ist die professionelle Lösung für den Atari ST. Einfach zu bedienen – und dennoch mächtig. Die richtige Software für Gestaltungsaufgaben von der Skizze bis zur DTP-Grafik. Zu einem fairen Preis.

Unverbindliche Preisempfehlung: 278,– DM

Für jeweils 10,– DM (Schein) erhalten Sie von uns zu beiden Programmen Testdisketten.

Datenbank. Assoziativ. Schnell.

Aus tausenden von Daten diejenigen zu finden, die man wirklich braucht, erfordert eine leistungsfähige Datenbank. Eine die schnell und einfach zu bedienen ist und 'denkt' wie wir. Assoziativ.

THEMADAT orientiert sich nicht am wörtlichen (sprich: buchstabengetreuen) Inhalt einzelner Felder, sondern an Begriffen, die den ganzen Datensatz beschreiben. Bildhaftes Denken bringt Sie zum Ziel.

Zur Suche nach einzelnen Datensätzen klicken Sie von jetzt an lediglich mit der Maus die gewünschten Themen an, und in kürzester Zeit erhalten Sie die gesuchten Datensätze. Einfach so.

Zu jedem Eintrag können Verweise auf beliebige extern gespeicherte Texte und Grafiken gespeichert werden, die die gesuchten Informationen ergänzen oder erläutern. Text- und Grafikeditor sind zu diesem Zweck bereits in THEMADAT integriert.

Unverbindliche Preisempfehlung: 248,– DM



SHIFT · UNTERER LAUTRUPWEG 8 · D-2390 FLENSBURG · ☎ (0461) 2 28 28 FAX: 1 70 50

SCHWEIZ: EDV-DIENSTLEISTUNGEN · STIFTG. GRÜNAU · ERLENSTR. 73 · 8805 RICHTERSWIL · ☎ (01) 784 89 47
NIEDERLANDE: MoPro · POSTBUS 2293 · 3500 GG UTRECHT · ☎ (030) 31 62 47
ÖSTERREICH: AMV-BÜROMASCHINEN · MARIAHILFERSTRASSE 77-79 · 1060 WIEN · ☎ (0222) 586 30 30

auf so eine Art und Weise, daß sich Ein- und Ausgabeaufrufe ohne explizite Angabe der Datei automatisch auf SysIn bzw. SysOut beziehen. An Ausgabeoperationen gibt es OutText, OutInt, OutReal, OutChar u.s.w., wobei die Bedeutung offensichtlich ist. Mit Hilfe dieser Operationen wird erst einmal nur fortlaufend in einen Zeilenpuffer Image geschrieben, der für jede Datei getrennt verwaltet wird. Sobald der Puffer voll ist, wird er erst tatsächlich ausgegeben. Vorher kann man aber die Ausgabe mit OutImage erzwingen. Der Aufruf von OutImage bei leerem Puffer bewirkt logischerweise die Ausgabe einer Leerzeile.

Eine andere Variante ist in der Funktionsprozedur IntInput (Zeile 7) zu bewundern: BreakOutimage gibt den Puffer aus so weit er gerade beschrieben ist und setzt die Schreibmarke auf dem Bildschirm unmittelbar dahinter. Dadurch lassen sich z.B. wie in diesem Fall Prompts realisieren.

Bleiben noch die Eingabeoperationen. Analog zur Ausgabe gibt es hier InText, InInt, InReal, InChar u.s.w.. In IntInput wird InInt verwendet (Zeile 9), das von der aktuellen Position im Puffer ab erst einmal alle Leerzeichen überliest, und wenn es dann immer noch nichts gefunden hat, eine neue Zeile von der Tastatur anfordert. Sobald die Funktion auf Ziffern trifft, wandelt sie diese in den entsprechenden Zahlenwert um und liefert diesen zurück. Mit InImage wird das Einlesen der nächsten Zeile erzwungen, auch wenn der Puffer noch nicht vollständig ausgelesen sein sollte. Die Funktionsprozedur TextInput ist so geschrieben, daß sie keinen Text, also die Konstante NoText, zurückliefert, falls die Datei zu Ende ist. Dies kann üblicherweise mit EndFile getestet werden, was aber bei der TOS-Konsole leider nicht funktioniert. Hier muß man sich damit behelfen, daß der Benutzer beispielsweise wie in diesem Fall ersatzweise "///" als Endekennzeichen eingibt. Die Abfrage erfolgt dann dadurch (Zeile 10), daß ein Teiltext des Eingabepuffers SysIn.Image gebildet und dieser mit der Konstanten "///" verglichen wird. Da sowohl SysIn als auch SysOut einen Puffer Image besitzen, muß die Dateireferenz zur Unterscheidung vorangestellt werden. Falls eine Eingabe erfolgte, liefert TextInput diese zurück. Dazu wird mit Strip der Teiltext des Puffers gebildet, der hinten keine Leerzeichen mehr enthält, und mit Copy kopiert.

Zum Schluß noch einige Anmerkungen zu den Prozeduren. SIMULA unterscheidet nicht zwischen Funktionen und Prozeduren, denn jede Prozedur kann durch Voranstellen eines Typnamens einen Typ erhalten und dann über ihren Namen einen Funktionswert zurückliefern. Umgekehrt kann eine Funktionsprozedur auch wie eine normale Prozedur verwendet werden. Deshalb sieht man häufig Prozeduren, die zusätzlich einen Wert liefern, und zwar um ihre Verwendung in Ausdrücken und somit eine elegantere Schreibweise zu ermöglichen. Ein typisches Beispiel dafür ist Upcase (SORT.SIM, Zeile 27). Bei Arrays als Parameter werden die Grenzen nicht angegeben. Deshalb kann auch - im Gegensatz zu Pascal - QuickSort vorübersetzt werden, ohne daß die Größe des Arrays bekannt sein muß. Die Grenzen der ersten Dimension können in der Prozedur mit LowerBound(a,1) bzw. mit UpperBound(a,1) ermittelt werden, was aber im Falle von QuickSort nicht benötigt wird.

Damit hätten wir eigentlich schon alle Besonderheiten von SIMULA vorgestellt, die nichts mit Simulation oder OOP zu tun haben. Sie sehen also, wenn Sie für den Anfang nur die Ihnen von anderen Sprachen her bekannten Konstrukte verwenden, dann ist das Programmieren in SIMULA zumindest genauso einfach und auf jeden Fall ein gutes Stück komfortabler. Und in dem Maße, wie Sie vertrauter mit SIMULA werden, können Sie sich nach und nach an komplizierteren Konstruktionen versuchen und so allmählich auch in die OOP einsteigen.

W.Rupflin

Einkaufsführer

Hier finden Sie Ihren Atari Fachhändler

1000 Berlin



HD Der ATARI ST Hardware Spezialist
Computertechnik
PUBLIC DOMAIN SERVICE
Sämtliche verfügbaren ST COMPUTER - PD's. vorrätig und weitere Serien für den ST, z.B. PD-Pool und ST-Vision.
1000 Berlin 65 · Pankstr. 42
Tel.: 030/465 70 28
SERVICE STATIONEN
Auch hier alle PD's vorrätig!
1/44, Lahnstrasse 94
1/20, Schönwalder Str. 65



1000 Berlin









1000 Berlin



Ihr Spezialist in Berlin für Hardware + Zubehör Eigenes Softwarestudio über 1000 verschiedene Titel am Lager

ATARI-Fachmarkt
NEC-Fachhandel · MS-DOS Fachmarkt
Katzbachstraße 6 + 8 · 1000 Berlin 61
☎ 030/7864340

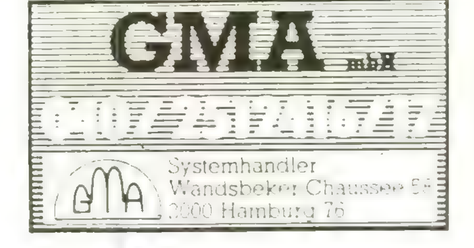




2000 Hamburg

Planen • Beraten • Realisieren
HCAG HABA COMPUTER AG
ATARI Fachhändler
Münsterstraße 9, D-2000 Hamburg 54
Telefon 040/56 19 09-0
Telefax 040/56 19 09-80





2000 Hamburg

DTP u. Werbeservice
Soft- und Hardwareversand
Viola Jaap
Berner Heerweg 512
2000 Hamburg 72
Telefon (0 40) 6 44 06 68

RADIX Bürotechnik
Heinrich-Barth-Straße 13
2000 Hamburg 13
Telefon (0 40) 44 16 95
NEU: Software Shop

2000 Norderstedt



2120 Lüneburg

Sienknecht
Bürokommunikation
Beratung - Verkauf - Werkstatt
Heiligengeiststr. 20, 2120 Lüneburg
Tel. 04131 / 46122, Btx 40 24 22
Mo.-Fr. 9^00-18^00 und Sa. 9^00-13^00

2210 Itzehoe



Coriansberg 2 · 2210 Itzehoe
Telefon (0 48 21) 33 90 / 91

2300 Kiel



MCC Computer GmbH Holzkoppelweg 19a
2300 Kiel 1 TEL 0431/54381 FAX 541717

2800 Bremen



Faulenstraße 48—52
2800 Bremen 1
Telefon (04 21) 17 05 77

2940 Wilhelmshaven

Radio Tiemann
ATARI-Systemfachhändler
Markstr. 52
2940 Wilhelmshaven
Telefon 0 44 21 - 2 61 45

3000 Hannover


DATALOGIC COMPUTERSYSTEME
ATARI ST- COMPUTER · HARDWARE · SOFTWARE
BERATUNG · SERVICE · VERKAUF
CALENBERGER STR. 26
3000 HANNOVER 1
TEL.: 0511 - 32 64 89

COM DATA
Am Schiffgraben 19 · 3000 Hannover 1
Telefon 05 11 - 32 67 36

3400 Göttingen

Büroeinrichtungs-Zentrum


3400 Göttingen-Weende
Wagenstieg 14 – Tel. 05 51 / 38 57-0

3500 Kassel

Hermann Fischer GmbH
autorisierter ATARI-Fachhändler
Rudolf-Schwander-Str. 5-13
3500 Kassel
Telefon (05 61) 70 00 00

4000 Düsseldorf

H O C O
EDV ANLAGEN GMBH
Ellerstraße 155
4000 Düsseldorf 1
Telefon 02 11 / 78 52 13



4000 Düsseldorf
Erkrather Straße 169-179
Tel. (0211) 73 62 14 • Fax 73 31 737

Hard und Software
Werner Wohlfahrtstätter
Atari
Public Domain
Atari Spiele
Atari Anwender
Ladenlokal
Irenenstraße 76c
4000 Düsseldorf-Unterrath
Telefon (02 11) 42 98 76

BERNSHAUS GmbH
Bürotechnik – Bürobedarf
Cäcilienstraße 2
4000 Düsseldorf 13 (Benrath)
Telefon 02 11 - 71 91 81

4010 Hilden

Beachten Sie unsere Anzeige in diesem Heft!


Computer · Drucker · Software · Bücher · Service
Gustav-Mahler-Straße 42-44
Tel. (0 21 03) 3 18 80 + 4 12 26

4150 Krefeld



DTP-CENTER
MATRIX *MatScreen*
ROLAND *Schnittplotter*
DATACOPY *Scanner*
Digitizer
C.O.P COMPUTER GMBH
Ennsstrasse 20
Postfach 507
4150 Krefeld 1
 (0 21 51) 54 20 91
Fax (0 21 51) 54 20 94

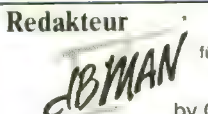
4150 Krefeld



4150 Krefeld
Ostwall 138
Tel. (0 21 51) 15 03 • Fax 69 409

4200 Oberhausen

Redakteur · Spectre 128



für TOS, DOS, UNIX, MAC, AMIGA und andere Systeme
by COMPUTER MAI

dBMAN-Vertretung für NRW und BENELUX
ISYS-COMPUTER GbR
Tel.: 02 08 / 65 50 31 · Telefax: 02 08 / 65 09 81
Max-Eyth-Straße 47 · 4200 Oberhausen 11

4300 Essen

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Limbecker Platz 4300 Essen 1
Tel.: (02 01) 17 63 99



4300 Essen
Limbecker Straße 12-16
Tel. (02 01) 23 35 15 · Fax 23 38 34

4320 Hattingen



4422 Ahaus

ATARI · Epson · Fujitsu
Molecular · NCR · Tandon · Schneider · Star

OCB

OCB-Computershop Wallstraße 3 4422 Ahaus Tel. 0 25 61/50 21	OCB-Hard- und Software Wessumerstraße 49 4422 Ahaus Tel. 0 25 61/50 21

4430 Steinfurt

CBS GmbH
COMPUTERSYSTEME
Tecklenburger Str. 27
4430 Steinfurt-Burgsteinfurt
☎ 02551/2555

4500 Osnabrück

Heinicke-Electronic
Meller Str. 43 • 4500 Osnabrück
Fax (0541) 58 66 14
Telefon (0541) 58 76 66
Wir liefern Micro-Computer seit 1978

4520 Melle

CBS GmbH
COMPUTERSYSTEME
4430 Steinfurt Tel. 02551/2555
Haferstraße 25 4520 Melle
Tel.: 05422/44788

4600 Dortmund

Elektronik
Computer
Fachliteratur
ATARI-System-Fachhändler
4600 Dortmund 1, Guntherstraße 75, Tel. (02 31) 57 22 84



city-elektronik

ATARI Systemfachhändler

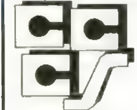


KARSTADT Aktiengesellschaft
Kampstraße 1 · 4600 Dortmund
Telefon (02 31) 5 43 91

Bei uns werben bringt
GEWINN
Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057
BUF

4600 Dortmund

cc Computer Studio GmbH



Atari-Systemfachhändler
PCs von Tandy Schneider Peacock
Drucker von Star Brother NEC
Elisabethstr. 5
4600 Dortmund 1
Tel. 0231/528184 Tx 822631 cccsd Fax 0231/528131

4650 Gelsenkirchen-Horst

MENTIS GmbH
Hard- und Software, Literatur
Bauteile, Service, Versand
Groß- und Einzelhandel
Poststraße 15 · 4650 Gelsenkirchen-Horst
Telefon (02 09) 5 25 72

Bei uns werben bringt
GEWINN



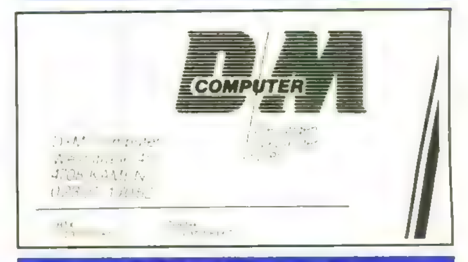
Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057
BUF

4650 Gelsenkirchen



Fest- und Wechselplatten (SCSI)
Großmonitore, Scanner, Software
ComputerSysteme und Anwendungen
Hüttenstraße 56 - 4650 Gelsenkirchen
Tel. (0209) 203420 FAX (0209) 271584

4708 Kamen



4800 Bielefeld

hardware
software
organisation
service



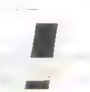
CSF COMPUTER & SOFTWARE GMBH
Heeper Straße 106 – 108
4800 Bielefeld 1
Tel. (05 21) 6 16 63

5090 Leverkusen

Rolf Rocke
Computer-Fachgeschäft
Auestraße 1
5090 Leverkusen 3
Telefon 0 21 71 / 26 24

5210 Troisdorf

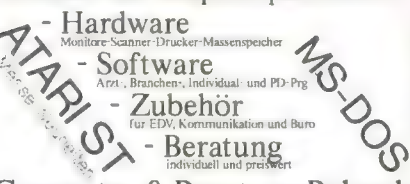
LOGITEAM
Computerhandelsgesellschaft mbH
Kölner Straße 132
5210 Troisdorf
Tel. (0 22 41) 7 18 97
FAX (0 22 41) 7 58 58



5253 Lindlar



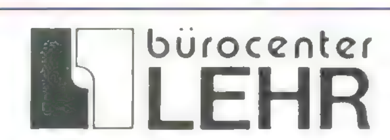
5300 Bonn



5430 Montabaur



5500 Trier



Güterstraße 82 · 5500 Trier
☎ 06 51 / 20 97 10
Fordern Sie unsere Zubehör-Liste an!

5600 Wuppertal

MEGABYTE
Computer Vertriebs GmbH

Friedrich-Engels-Allee 162
5600 Wuppertal 2 (Barmen)
Telefon (02 02) 8 19 17

5630 Remscheid

COM SOFT
Nordstraße 57 · 5630 Remscheid
Telefon (0 21 91) 2 10 33

5650 Solingen

MegaTeam
Computer-Vertriebs-OHG
Kölbach – Finke
Hardware – Software – Zubehör – Service
Rathausstraße 1–3 · 5650 Solingen 1
Telefon (02 12) 4 58 88 · Fax (02 12) 4 73 99

5800 Hagen



Vertragshändler Axel Böckem
Computer + Textsysteme
Eilper Str. 60 (Eilpezentrum) · 5800 Hagen
Telefon (0 23 31) 7 34 90

6000 Frankfurt

WAIZENEGGER
Büroeinrichtungen
Kaiserstraße 41
6000 Frankfurt/Main
Tel. (0 69) 27 30 6 - 0

Eickmann Computer

***Der* Atari-System-Fachhändler !**

z.B.: Festplatten von 30 MB bis 110 MB für Atari ST und Mega ST, Zusatzaufrüstungen für Ihre Festplatten bis 110 MB, Umrüstung Ihres SM 124 in einen EM 124 Multisync für alle Auflösungen, Slotkit für PC 1...

besuchen Sie unser Fachgeschäft:
In der Römerstadt 249
6000 Frankfurt 90-Praunheim
Telefon (069) 763409

6100 Darmstadt

Heim
Büro- und Computermarkt
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon (0 61 51) 5 60 57

6108 Weiterstadt



6123 Bad König



! Vor(Ver)führungen !

CALAMUS	OUTLINE ART	ARABESQUE
Repro Studio ST	PEGASUS +	DRAFTER CAD
TmS Graphics	TmS CRANACH	TmS Vektor
Sherlook OCR	SIGNUM ! 2	SCRIPT
ADIMENS PLUS	BSS Plus	Reprok
Eickmann Turbo 16 MHz		PC SPEED 14
PANASONIC Scanner 506 U	LACOM Wechselplatte	

Speichererweiterung 2 MB: 598.– / 4 MB: 998.–
Laufwerk 5 1/4" 40/80 Track: 298.–
Hard und Software aller Hersteller

6200 Wiesbaden



Bei uns werben bringt

GEWINN

Sprechen Sie mit uns.
Heim Verlag ☎ 0 61 51 / 5 60 57
BUF



COMPUTERCENTER
DER SYNTHESIZER • STUDIO JACOB GMBH
Computer, Software, Zubehör
MIDI - Equipment, eigener Service
6200 WIESBADEN, MAINZER STR. 137
Tel. (0 61 21) 71 94 90 (0 61 21) 71 96 45

6236 Eschborn



6240 Königstein

KFC COMPUTERSYSTEME
Wiesenstraße 18
6240 Königstein
Tel. 0 61 74 - 30 33
Mail-Box 0 61 74 - 53 55

6250 Limburg



DTP-Center • Salzgasse 6 • 6250 Limburg
☎ 06431/5004-0 • FAX 06431/5004-10

6300 Gießen



6400 Fulda

Schneider ATARI Commodore
WEINRICH
BÜRO · ORGANISATION
Ronsbachstraße 32 · 6400 Fulda
Telefon (06 61) 4 92 - 0

6457 Maintal

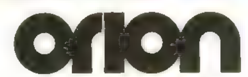


6457 Maintal-Dörnigheim Robert-Bosch-Straße 14
Tel. (06181) 4 52 93 Fax (06181) 43 10 43
Mailbox (06181) 4 88 84 Btx *2 98 99#

6500 Mainz - Hechtsheim



6520 Worms



Computersysteme
GmbH
6520 Worms · Friedrichstraße 22
Telefon 0 62 41 / 67 57 - 58

6581 Niederbrombach

INFODAT GmbH
ATARI Computersysteme
Protar Center
Beratung - Softwareentwicklung - Individuallösungen
Unter Hochkastell 3 *Telefon:*
6581 Niederbrombach *(06787) 1425*

6630 Saarlouis



6700 Ludwigshafen

MKV Computermarkt
Bismarck-Zentrum
6700 Ludwigshafen
Telefon 06 21 - 52 55 96

6720 Speyer

THEILLE
Computersysteme
Gilgenstraße 4 · 6720 Speyer
Telefon (0 62 32) 772 16



Die Fachleute für Computer
Autorisierter ATARI - Vertragshändler
Hardware, Software & Zubehör für
• Apple • Acorn • Jonathan • MS/Dos •
6720 Speyer Auestraße 20
Telefon 06232 /32428 oder 32435
Mo.-Fr. 9 - 12 u. 15 - 18.30, Sa. 9 - 14 Uhr

6750 Kaiserslautern

INFODAT GmbH
ATARI Computersysteme
Protar Center
Beratung - Softwareentwicklung - Individuallösungen
Schubertstr. 16 6750 Kaiserslautern
Tel: (0631) 63597 Fax: (0631) 63589

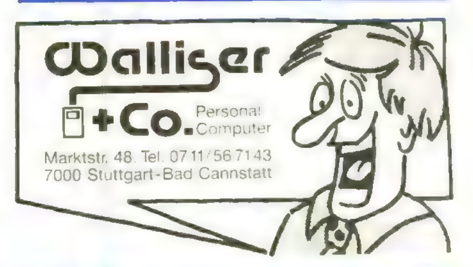
6800 Mannheim

GAUCH + STURM
Computersysteme + Textsysteme
6800 Mannheim 24
Casterfeldstraße 74-76
☎ (06 21) 85 00 40 · Teletex 6 211 912

6900 Heidelberg

JACOM FAMILA-CENTER
Hardware · Software
Schulung · Service
Hertzstraße 1 · 6900 Heidelberg 1
Telefon (0 62 21) 30 00 44-5

7000 Stuttgart



7047 Jettingen

Verkauf - Service - Software
Norbert Hlawinka
Heilbergstraße 3
Im Multi-Center
7047 Jettingen
Telefon (0 74 52) 7 76 15



COMPUTER SHOP

7100 Heilbronn



Computer-Welt
Am Wollhaus 6
7100 Heilbronn
Tel. 0 71 31 - 6 84 01 - 02

7100 Heilbronn



7150 Backnang



7312 Kirchheim/Teck

Comp & Phone
Computerfachhandel
Alleenstraße 66
7312 Kirchheim/Teck
Tel.: 07021/3949 - Fax: 07021/53933

7410 Reutlingen

MKV GMBH

Listplatz 2
7410 Reutlingen
Telefon 0 71 21 - 3 66 47

7475 Meßstetten

Ihr ATARI-Systemhändler im Zollern-Alb-Kreis
HEIM + PC-COMPUTERMARKT
HARDWARE · SOFTWARE · LITERATUR
SCHEURER
ATARI COMMODORE CUMANA DATA BECKER
MULTITECH RITEMAN SCHNEIDER THOMSON
7475 Meßstetten 1 Hauptstraße 10 0 74 31 / 6 12 80

7500 Karlsruhe

MKV GMBH

Kriegsstraße 77
7500 Karlsruhe
Telefon (07 21) 8 46 13

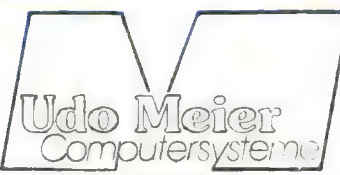


7600 Offenburg

FRANK LEONHARDT ELECTRONIC
Ihr Fachgeschäft für Microcomputer · Hifi · Funk

In der Jeuch 3
7600 Offenburg
Telefon 07 81 / 5 79 74

7700 Singen



Ringstraße 4
Telefon (0 77 31) 6 82 22

7730 VS-Schwenningen

BUS BRAUCH & SAUTER COMPUTER TECHNIK

Villinger Straße 85
7730 VS-Schwenningen
Telefon 0 77 20 / 3 80 71-72

7750 Konstanz

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Rheingutstr. 1 · ☎ 0 75 31-2 18 32

7890 Waldshut-Tiengen



service gmbh
Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon 0 77 51 / 30 94

7910 Neu - Ulm



7918 Illertissen

bioTech gmbh
technische Informationssysteme
Computerladen

Marktplatz 13
7918 Illertissen
0 73 03 / 50 45

7930 Ehingen



8000 München

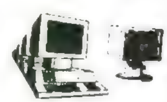


COMPUTER + BÜROTECHNIK
COMPUTER · SOFTWARE · PERIPHERIE
BERATUNG · TECHN. KUNDENDIENST
INGOLSTÄDTER STRASSE 62L
EURO INDUSTRIE-PARK · 8000 MÜNCHEN 45
TELEFON 089 / 31 81 95-0 · TELETEX 89 83 41

City Studio
Rindermarkt 6, 8000 München 2
Tel. 089 / 2 60 98 01

schulz computer
Schillerstraße 22
8000 München 2
Telefon (0 89) 59 73 39
Beratung · Verkauf · Kundendienst

T.S. Service
Szenere Hard & Software
Schleißheimerstr. 220
D-8000 München 40



Tel. 089/3089408 Fax 089/3085636 ATARI
Festplatten / Rechner / Zubehör / Service
Bürozeiten: Mo. bis Fr. 9.00-12.00 13.00-18.30 Samstag 9.00-14.00

8032 Gräfelfing

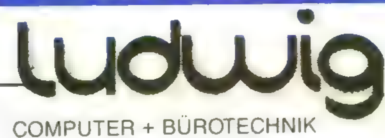


Pasinger Straße 94 8032 Gräfelfing
Telefon 089 - 8548823
Fax 089 - 8541764

8150 Holzkirchen



8200 Rosenheim



COMPUTER + BÜROTECHNIK
Kufsteiner Str. 11, 8200 Rosenheim
Tel. (08031) 38 00 30
Fax (08031) 1 53 34

8390 Passau



8400 Regensburg



8423 Abensberg

WITTICH
COMPUTER GMBH
Tulpenstr. 16 · 8423 Abensberg
☎ 0 94 43 / 4 53



8500 Nürnberg



8500 Nürnberg



8520 Erlangen

Computerservice Decker
Meisenweg 29 - 8520 Erlangen
Telefon 0 91 31 / 4 20 76



8600 Bamberg



Bei uns werben bringt
GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057
BUF

Computer und mehr...
PC
8600 Bamberg
Ludwigstraße 2
Tel. (0951) 2 51 37-38 • Fax 2 63 94

8700 Würzburg

SCHOLL
BÜROTEAM
Hardware · Software
Service · Schulung
computer center
am Dominikanerplatz
Ruf (09 31) 3 08 08 - 0

8720 Schweinfurt

Uhlenhuth GmbH
Computer + Unterhaltungselektronik
Albrecht-Dürer-Platz 2
8720 Schweinfurt
Telefon 0 97 21 / 65 21 54

8900 Augsburg



Unser Plus: Beratung u. Service
Schwalbenstr. 1 · 8900 Augsburg-Pfersee
Telefon (08 21) 52 85 33 oder 52 80 87
Computer Vertriebs- und Software GmbH



8906 Gersthofen



8940 Memmingen



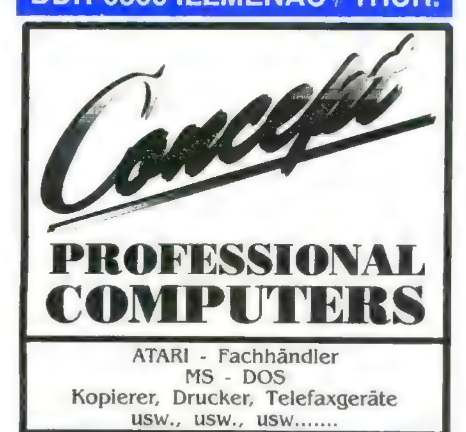
8960 Kempten



DDR-5800 GOTHA



DDR-6300 ILLMENAU / THÜR.



Porzellanstraße 30 · 6300 Ilmenau/Thür. · Tel. 2965

DDR-8027 DRESDEN

Dresdner Computer Center

Ihr erster ATARI System-Fachhändler in Dresden

Kaitzerstraße 82
8027 Dresden Tel.: 47 88 65

ÖSTERREICH

A-1030 Wien

Ihr ST-Fachhändler in Wien

Computer-Studio

Wehsner Gesellschaft m.b.H

A-1030 Wien
Landstraßer Hauptstraße 2
Hilton-Einkaufspassage

A-1040 Wien

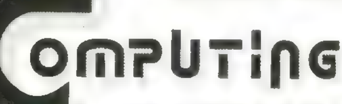
Ihr ST-Fachhändler in Wien

Computer-Studio

A-1040 Wien · Paniglgasse 18-20
Tel. (02 22) 505 78 08, 505 88 93

A-1180 Wien

Ihr kompetentes Atari-Fachgeschäft in Wien



Tel. (0222) 408 52 56
Fax: (0222) 408 99 78
A-1180 Wien - Schulgasse 63

SCHWEIZ



Ihr Computer Spezialist

5000 Aarau, Bahnhofstrasse 86, Tel. 064/22 78 40
4102 Basel-Binningen, Kronenplatz, Tel. 061/47 88 64
5430 Wettingen, Zentralstrasse 93, Tel. 056/27 16 60
8400 Winterthur, St. Gallerstrasse 41, Tel. 052/27 96 96
8021 Zürich, Langstrasse 31, Tel. 01/241 73 73

Grösste Auswahl an Peripherie, Software, Literatur und Zubehör.

CH-1205 Geneve

PIMENT ROUGE
INFORMATIQUE S.A.
8, RUE DES MARAICHERS
1205 GENEVE TEL. 022/28 56 24

CH-1700 Fribourg

FRIDAT SA INFORMATIQUE
ehem. Softy Hard's Computershop

VOTRE SPECIALISTE
Rte des Grives 4
1700 Granges-Paccot/Fribourg
Tel. 0041 (0)37 26 66 28
Fax. 0041 (0)37 26 61 06

CH-2503 Biel

UE URWA ELECTRONIC
Computer Hard- und Software

Ihr ATARI ST Spezialist
in der Schweiz.
☎ 032/41 35 35

Bözingenstraße 133, 2504 Biel

CH-3072 Ostermundingen



CH-4313 Möhlin

BCR Computerdienst
Bahnhofstrasse 63
CH-4313 Möhlin

Computersysteme
EDV-Beratung
Installationen
CAD Anlagen
Datenpflege + Service

ATARI
NEC



Tel. 061 88 30 32 FAX 061 88 30 03

CH-4500 Solothurn

Fluri Informatik

Hard- & Software, Zubehör
ATARI Schulungszentrum
Desktop Publishing
Systemlösungen

Schänzlistr. 4 4500 SOLOTHURN 1
Tel. 065 / 23 68 58 Fax. 065 / 23 16 57

SCHWEIZ

CH-4625 Oberbuchsiten

STECTRONIC M. Steck
Electronic-Computer-Shop

Hauptstr. 104/137
CH-4625 OBERBUCHSITEN
Tel. 0 62/63 17 27 + 63 10 27

CH-4665 OFTRINGEN

Warum nicht gleich zum Fachmann ?

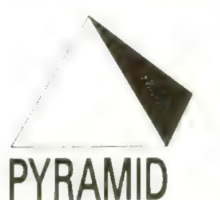
IHR ATARI PARTNER

Unser Computerzubehör

Drucker	von	Panasonic, Epson
Plotter	von	HP, Seikosha
Monitore	von	Epson, Matrix, Sony
Scanner	von	Panasonic, Epson,
Festplatten	von	Atari, Vortex,
Belichter	von	Ultre

Unsere Dienstleistungen

Beratung	Verkauf
Installation	Schulung
Service	Reparatur
Scann	Druckservice
Plott	Belichtungsservice



PYRAMID
PYRAMID electronics AG
Industrie Kleinfeld 33
4665 Oftringen
Tel: 062/97 33 55 Fax: 062/97 30 79

LUXEMBURG



****

* R A M - Erweiterungen spottbillig ... *

****

* S I P s' jetzt für a l l e S T s '!! *

****

* I. für 520er/ 1040er
* z.B. 4MB=799,--(inkl.Einbau) *

****

* Letzte Neuheit ist die S I P -
* Platine. Gewaltig winzig
* zig. 4 MB auf 5x8 cm !!
* Platzprobleme mit anderen
* Erweiterungen? Wie bitte ?!

****

* Bestückt f.2.5 MB 475.- *

****

* Der Clou: Irgendwann 2 wei-
* tere SIPs' einstecken = 4MB
* 1 SIP bei mir = 175,-

****

* oder gleich: Mit
* 4 SIPs' (=4MB) 799.- *

****

* Der Einbau ist inklusive. *

****

* II. A T A R I S T / E
* 2.5 MB ab 290,-- !! *

****

* Wie? Ich "tausche" 2 große
* (MB-) SIMs' gegen 3 von den
* 4 Kleinen, die im Original-
* ST/E sind. Plus die 290,--
* Ohne Umtausch kassiere ich
* 165,-/MB SIM. Macht 330,-
* Alles klar?Nicht?? Anrufen! *

****

* MEGA-2 auf 4 MB = 475,--
* **MEGA1 auf 2 MB ab 375,-** *

****

* Fragen ... Fragen .. Fragen
* - Welche 4 (2.5)MB Erweite-
* rung sonst ist kürzer als
* eine Zigarette, schmaler als
* eine Packung Streichhölzer ? *
* Fragen? Anrufen !

****

* **Nicht-Hamburger** sollten
* 12 DM hinzurechnen :
* -ST als Wertpaket -wird fix
* und gut behandelt- zu mir,
* macht 12,- plus die 16,- f.
* NN u.Rückversand. Immer
noch echt billig, näh? (näh?=was
* =wa! = eh!!= hoast mi ?!?)

****

* Und immer noch: 24-Stunden *
* Umbau nach Terminabsprache*
* 6 Monate Garantie.

****

* Sven Betz - 2 Hamburg 20
* Hohe Weide 50
* Tel. 040-420 43 63

****

Kleinanzeigen

Verschiedenes

Farbmonitor Atari DM 400,- Tempus 1.01 90,- DB-MAN 5.1 800,- fibuMAN e 300,- fibuMAN f 4.0 650,- wegen Nichtgebrauchs zu verkaufen. C. HEITZ Sinabronnerstr. 33 D-7901 LONSEE 0 73 36 / 56 05 FAX 56 17

***** Scan-Studio Kluge *****
Wir scannen jede Vorlage bis DIN-A4 (Foto, Grafik, Text). je DM 3,-!! Höchste Auflösung! Texterkennung, Laserdruck, Beratung! Tel.: 0 24 51 / 6 81 79

Inhaltsverzeichnis 6/88-6/90 ST-Computer und ST-Magazin auf Disk zusammen DM 20, Programm hierzu kostenlos. Superschnell! Bestellung (Vork.) F. Schott, Bahnhofsplatz 3, 8425 Neustadt/Donau.

Private Computer Börse
Vermittelt jede Hard-Software bei **Kauf, Verkauf, Tausch**
Info: Tel.: 0 73 71 / 71 67
v. Mo.-Fr. 17 - 20 Uhr, Sa. 9 - 12 Uhr

SUPER! LARRY-3-Lös. u.a.! DM 20 an Hervol, Gabrielenstr. 2, 8000 Mü 19

Farbbänder f. (fast) alle Drucker, Etiketten in vielen Formaten (400 St. 89*36 mm DM 7,50) kostenl. Infomappe anfordern: W.Kuhn EDV-Zubehör, Hessenstr., 6340 Dillenburg 2, 0 27 71 / 3 26 88

Kleinanzeigen

Biete Hardware

Speichererweiterung ST 260/520
2,5 MB: DM 450,- 4 MB: DM 880,-
Einbau +10,-, mit Garantie
heiler engineering 07 21 / 37 66 22

1-MB-Erweiterung
für 260/520 ST, mit Einbau,
Huckepackgelötet, 150,- DM
0 71 95 / 6 76 70, ab 20 Uhr

0,5-MB-SPEICHERERW.: 0431569216

2,5 Megabyte Platine für Atari
ST (außer STE) DM 79 + NN 7,50
Bausatz! Info gratis 0408300199

Sound Sampler 110 DM, Hardware
Uhr 29 DM, TV Modulator 98 DM
* Kostenloses Info anfordern *
Electronic Service Roslawski
Amtsstraße 2a, 4352 Herten

1MB, ROM-TOS 260ST; SF 314; MAUS
ADIMENS 2.1; 600 DM; 02 08/60 13 29

Mega ST 4, Monitor, NEC P7
mit autom. Einzelblatteinzug: Wegen
Auslandsstudium zu verkaufen!
Freiburg Br. Tel: 07 61 / 3 73 52

AMSTRAD 386 20MHz 65MBHD
VGA COLOR 14" 4MB RAM 2WO-
CHEN ALT FP: 5000,- 0 99 41 / 43 63

Atari-Rainbow-ROM-TOS 1.4
mit IBM-Grafikzeichen anstelle des
hebräischen Alphabetes und ST-Patches,
auf 6 Eproms 90 DM, auf 2 Eproms mit
Adaptersockel 120 DM
Tel. 0 26 30 / 75 25

TOS 1.4 MIT ST-COMP-PATCHES
IN 6 EPROMS 65 DM BEI VOR-
KASSE, SONST + NACHNAHME:
POSTFACH 3345, 2900 OLDENBURG

Mega-Tower (01/90) 2,5(4 MB), 85
MB SCSI im Wechselrahmen, 2/3 LW
Umsch., 2*1.44 MB LW, Blitter,
Monitor Umsch., Außerdem 2*720
KB LW, RTS-Tastenk. für 1040 ST
Tel.: 0 64 02 / 28 17, tgl. ab 19 Uhr

Maus NEU 45 DM Tel. 0 80 91 / 40 15

Maxon Grafik-Karte MGE II
für Mega-ST, Auflösung frei, Prozessor
i82786, max. Aufl. 1280*1024,
13,75-110 MHz, max 256 Farben aus
262144, 2150,- Tel.: 0 20 52 / 8 01 95

ATARI 520ST+ (1MB), WEIDE-UHR
2*720 KB-FLOPPY, SM124, MAUS
für 1200,- DM, Tel.0 22 51 / 5 75 00

Verkaufe: 1040 STF, NEC P6, Zubehör
VB 1800,- event. einzeln
Tel.: 0 86 69 / 3 71 50, ab 20 Uhr

I/O-INTERFACE ITF-32
TTL-komp. 32 Ein- u. Ausgänge,
erweiterbar f. Alarmanlage, Ind.
Steuerungen J. Sieber, Raffeltersteige 15
7100 Heilbronn Tel.: 0 71 31 / 16 11 51

520ST 1MB, 2*1MB Floppy,
30MB Festpl., Drucker, SM 124,
Farb-Monitor, Modem,
ca. 100 Disk. (Spiele/Anwend)
umständehalber zu verk. VB 2200,-
Telefon: 07 11 / 37 86 91, ab 19 Uhr

RECHNERKOPPLUNG
ATARI ST - SHARP POCKETCOM.
NEUWERTIG DM 80
0 40 / 715 65 12

ATARI ST-Computersystem
520 ST (1MB, ROM-TOS), SM 124,
SF 354, SH 204, 5,25"-Laufwerk
inkl. Software, Leerdisketten
und Literatur: 1200,- SFr.
Tel.: (CH) 0 61 / 99 25 83, 19-21 Uhr

Festplatte SCSI 85 MB im MegaLook,
6 Monate für VB DM 880,-
Axel Braukmann Tel: 0 21 59 / 18 99

ATARI **MEGA ST4** inkl. SM 124,
10 Monate alt, für VB **2400,-** DM,
Axel Braukmann Tel: 0 21 59 / 18 99

---

* **SCSI Festplatten 100 % komp.** *
* 85 MB-28ms DM 1358,- / 80MB- *
* 24ms DM 1458,- / 49MB-40ms *
* DM 1158,- / 32MB-40ms DM 968,- *
* **Disketten 2DD 10 Pack DM 16,-** *
* SPE-TECHNIK WN 07151/5002-12 *

NEC P6-Nadeln. Tel.: 05 51 / 2 29 03

VERK.GEB. MAUS (50,-): 0431569216

Speichererweiterung ATARI-ST:
inkl. Einb/Gar. 1 MB nur 150,-
2.5 MB ab 450,- Tel.: 0 66 93 / 85 01

Juniorprommer 130,-, 5 EPROMs 27512
50,-, 1MB-Floppy 130,-
Org. SIGNUM2! inkl. Fonts 250,-
Tel.: 0 74 22 / 42 36

5,25" int. f. Atari? Wir haben es:
Shugart-Floppy 720/880 KB
(Keine HD-Disketten mehr erforderlich!),
40/80 Track incl. Anleitung DM 149,-
Mentis GmbH, Poststr. 15,
4650 Gelsenkirchen, Tel. 02 09 / 5 25 72

Biete Software

Achtung Lotto-Freunde!
Alle Lottozahlen auf Diskette
kostenloses Info anfordern.
U. Barsch Pf 1109, 7120 Bietigheim

G-I-R Software Vertrieb
Peter Grantz, Hauptstr. 49
D-2401 Ratekau/Lübeck

PD-Software für Atari ST,
- über 300 Disk
- Angebot wird lfd. erweitert.
- Update Service
- 24-Std.-Auftragsdienst
gewährleistet durch
Tel.: 0 45 04 / 41 15
BTX : 0 45 04 / 52 67

PD-Software, pro Disk. Fr. 5,-
Versand innerh. 24 Std.
Schweiz Tel.: 0 71 / 63 65 68

Ausschreibung-Angebot-Rechnung
Datanorm/Baustoffverwaltung
K-Zahlberechnung-Wärmebedarf
Wärmeschutz-Dampfd./Tauwasser
Dipl. Ing. V. Koch, Germersweg 21
3563 Dautphetal 3, Tel.: 0 64 68 / 76 52

ORIGINALE FÜR ATARI-ST,
oft weniger als der halbe Preis!!!
1ST.ADRESS 64,50, OPTIMIZER
79,90, PUBLISHING PARTNER + ST
PAINT 185,-, LATTICE C 3.04 128,-
MEGAMAX C 1.3 125,-, TIME
WORKS 1.12 165,- u.A. T.: 04191/4320

GFA 3.5 EWS ORIGINAL 180,- DM
Tel.: 0 21 73 / 7 78 17, nach 18 Uhr

Calamus Buch 60,-, PictureDisks I 90,-,
ST Plot 50,-, ST Math 70,-, TKC
Einnahme/Üb. 100,-, Spielesammlung
ST 30,-, Tel.: 0 69 / 507 16 94

Kompl. BSS PLUS System wegen
Systemwechsel 30 % unter NP abzu-
geben. Tel.: 0 87 43 / 24 75, ab 18 Uhr

GEBURTSHOROSKOPE!
Exakte Zeichnung, Aspekte,
Berechnungsdaten.
Interpretation für gewerbl. Einsatz!
Tel.: 0 61 51 / 2 37 20

#################################
ST-COMPUTER-PD-Disks je 3DM!
Versandkosten: VK 5DM / NN 7DM
Katalogdisk 3DM bei S. Heigert
Berndesallee 6, 6501 Heidesheim
#################################

PD zu Niedrigstpreisen!
ST-Comp., 2000er, eigene ab 1,95 DM
inkl. 2S/2D-Disk
Liste g. frank. Rückumschlag
Hintermeier, Pfingstweide 3
3551 Lahntal-Sarnau

ST **!Public Domain Software!** Auch
ABO! Bespielte 2DD 3,5" Disk ab
2,40 DM. Katalog gratis.
Ollis PD-Versand, Goethestr. 6,
6702 Bad Dürkheim. – Die Nr.1 –

Anwender- u. Spielesoftware
Laufend n. Programme a. Lager
Riesenauswahl an Original
Soft- u. Hardware
Gratiskatalog anf. o. anrufen
W. Wünsch
Soft- u. Hardware,
0 72 31 / 76 65 95
Friedenstr. 212, 7530 Pforzheim

######## PD-SOFTWARE ########
auch SIGNUM- und GRAFIK-Pakete
KATALOG-DISK 3DM / Kurz-INFO
0 DM bei: B. Jürgensen / Holmberg 4
2398 Harrislee / v. 8.-30.9. Urlaub

**** PD Software Atari ST ****
Je Set 5 Disks (2DD) für 30 DM
Set 1: Adventures, Strategie
Set 2: 21 Actiongames Teil 1
Set 3: 14 Actiongames Teil 2
Set 4: über 40 Anwendungsprog.
Set 5: 20 CAD, Mal, Animat.prg
Set 6: best of Music + Midi
Schnellversand Oliver Witzig
Himmerreichstr. 23, 7000 Stgt. 1
VK oder NN / 24 h / T.: 07 11 / 69 06 29

GFA Assembler	99,- DM
GFA Basic 2.0	29,- DM
GFA Basic 3.5	199,- DM
GFA Buch Trainer	19,- DM
GFA Buch Anwender	45,- DM
PD-Pool 2001-2120	240,- DM

Wegen Systemwechsel zu verk.
J. Weber, Randerathweg 56
4044 Kaarst

DER SCAN-DAL IST PERFEKT!
Bilder aus Programmen holen
und durch eigene ersetzen!
Für DM 39 bei: U. Witte, Am Born 13,
2000 Hamburg 50

1ST WORD PLUS 3.15 für DM 180
COMPUTER COLLEG für DM 100
Tel.: 05 51 / 79 11 68

********* **Hallo PD-User** *********
PD-Software ab 2.60 DM -jeden Monat
ein Topangebotspaket (10 Disk) 26,-
Abo ab 26 DM Katalog 2,50 DM
Bei Neumann Stahlwerkstr. 26,
4600 Dortmund

Grafiken u. Bilder für Signum u. STAD,
ca. 1500 Stk. inkl.
Katalog u. 3 Disketten DM 25,-
Tel.: 0 40 / 711 93 55

Standardlösungen sind gut
Branchenlösungen sind besser

Darum für **Malermeister**
von **Malermeistern**

MALER ST / / / / / / / / / / /
/ / / / / Der Malercomputer

Aufmaß, Angebot, Kalkulation
Baustellenüberwachung, uvam.

Weitere Infos bei
Malermeister, techn. Fachwirt
Thomas Henn, 7107 Neckarsulm
Heilbronner Str. 50
Tel.: 0 71 32 / 23 62 Fax: 62 64

SOFTWAREPROBLEME??
Programmiere ST-Soft nach Maß
Preiswert! Wenn möglich Raum Basel:
D. Grandchamp, Loogstr. 26, CH-4142
Münchenstein, Schweiz
Tel.: 0 61 / 46 41 42 – Es lohnt sich!!

---

* **Preissenkung!!!** *
* Softstation *
* 2 fortl. PD von ST-Comp. (1/2, usw) *
* oder 1 PD ab Nr 225 **DM 4,-** *
* bel. komp. und sonst. Serien *
* DM 5,- *
* Disk. wahlw. blau, weiß, rot, grün, *
* gelb od. orange *
* Porto + NN DM 6,- *
* Tel.: 0 71 95 / 5 37 07 *

Public-Domain-Software
ST-Computer, GFA, XEST usw.
je 3,40 DM inkl. 2S/2D-Disk
Lieferung innerh. 24 Stunden!
Liste auf Disk lieferbar
Tel.: 0 40 / 712 54 97

MODELLBAHNSTEUERUNG
MIT DEM ST mit Fahrplänen oder
direkt im Gleisbild, Fahrstr., Anfahr-
verz., Ansteuerung im Interrupt...
Info bei: Martin Birn,
Knoblochstr. 51, 7100 Heilbronn

Suche Software

Gebrauchte **Original**-Soft mit
Anleitung! (und Verpackung).
F. Schumann, Im Riegel,
6702 Bad Dürkheim 5

Suche Hardware

Suche günstig ST1096N und ICD-
Hostadapter „ADVANTAGE"
Tel.: 020543949

Tausch

Ich tausche PD 3:1 oder kaufe
PD Programme die **NICHT**
auf Deutsch oder Englisch
geschrieben sind.
Aktuelle PD aus USA.
JAMES MOHR 1033 JACQUELINE
WAY SAN JOSE CALIF 95129 USA

PD-Tauschpartner gesucht.
Baade, Forststr. 120, 4950 Minden.

Kontakte

Welcher Tischlermeister
arbeitet mit einem Atari ST
und würde mir bei meiner
buchhalterischen Arbeit helfen?
Hans-Fr. Klüssendorf, 3031 Magdeburg,
Zillestr. 2, Tel.: 0037 91391025

Programmier Praxis

INHALT

GREP

STRING-SUCHE IN MEHREREN DATEIEN

Jan Bolt

Während der Programmerstellung ist es häufig nötig, Zeichenketten im Quelltext zu suchen (Variablen, Funktionsnamen etc.). Solange das Projekt aus einer Datei besteht, erledigt das jeder Editor. Was aber, wenn 10, 20 oder 50 Dateien zu durchsuchen sind ? Hier tritt grep auf den Plan.

Grep ist ein Standard-Tool unter UNIX, das Dateien nach einer Zeichenkette durchsucht. Diese Implementation für den ST ist eine bescheidene Untermenge des Originals. Der Unterschied besteht hauptsächlich darin, daß der ST-grep nur feste Strings sucht, während sein großer Bruder eine mächtige Wildcard-Syntax erlaubt. Was kann nun unser grep? Der Aufruf erfolgt in der Form

```
grep -[cdhilnv] String Dateien
[>[>]output]
```

Gesucht wird dann nach String in Dateien. Es können 1...10 Dateien im Gemdos-Format angegeben werden. Pfade und Wildcards sind erlaubt. Wird keine Datei angegeben, liest grep von stdin (Tastatur), was im allgemeinen nicht viel Sinn macht. String kann alle Zeichen enthalten, die über sich die Tastatur eingeben lassen. Sollen Leerzeichen enthalten sein, muß die Zeichenkette in " oder ' eingeschlossen sein. " oder ' in String müssen durch das jeweils andere Anführungszeichen geschützt werden. Grep ist natürlich ein klassisches Tool für Kommando-Shells. Für den Aufruf vom Desktop habe ich Option -d (warte vor Programmende auf Taste) und Output-Redirection implementiert.

- \> Datei lenkt die Ausgabe in 'Datei' um.
- » Datei fügt Ausgabe an 'Datei' an.

Das Ausgabeformat ist normalerweise Dateiname:Zeile, in der String gefunden wurde

Die Optionen haben folgende Bedeutung:

- -c zeige lediglich Anzahl der gefundenen Zeilen
- -h keine Dateinamen anzeigen
- -i ignoriere Groß/Klein
- -l liste lediglich die Namen der Dateien, die String enthalten
- -n Ausgabe jeder gefundenen Zeile mit Zeilennummer
- -v Invertierung der Suche, d.h. alle Zeilen, die String nicht enthalten, werden berücksichtigt

Beispiele für den Aufruf:

```
grep -i move.w *.s
grep "Jan Bolt" *.c *.s
include\*.h >grep.log
grep -in 'string = "String";' *.c
```

Rückgabe an das aufrufende Programm: 0 Zeile(n) wurde gefunden 1 keine Zeile wurde gefunden 2 Fehler ist aufgetreten

Zum Programm:

Das Hauptprogramm main erzeugt aus der Kommandozeile die üblichen Parameter int argc (Anzahl der Argumente), char *argv[] (Zeiger auf ein Array mit den Adressen der Argumente). Die vom C-Startmodul übergebenen Parameter können nicht benutzt werden, weil diese in der Annahme Leerzeichen = Argumenttrenner erzeugt wurden. Die Funktion do_main ist das eigentliche Hauptprogramm und interpretiert die Kommandozeile. do_grep übernimmt das Handling der Dateien und ruft für jede gefundene Datei do_file auf. Hier wird nun jede Zeile daraufhin überprüft, ob der Such-String enthalten ist, und ggf. ausgegeben. Diese Überprüfung findet in der Funktion get_match statt, die z.Z. lediglich einen Aufruf von strstr enthält (Suchen von festen Strings). Vielleicht hat ein Leser dieses Beitrags Lust, die Funktion zu erweitern.

Literatur:
UNIX manual, Text Utilities



```
  1:  /*============================================*
  2:   *                                            *
  3:   * grep.c                      Version 150590 *
  4:   *                                            *
  5:   * sucht String in Textdateien                *
  6:   *                                            *
  7:   * Aufruf: grep [-cdhilnv] String Dateien     *
  8:   * [>[>]output]                               *
  9:   * -c zeige nur Anzahl der gefundenen Zeilen  *
 10:   * -d warte vor Programmende auf Taste        *
 11:   * -h keine Dateinamen anzeigen               *
 12:   * -i ignoriere GroP/Klein                    *
 13:   * -l liste Dateien, die String enthalten     *
 14:   * -n Ausgabe jeder gefundenen Zeile mit Zeilnr *
 15:   * -v liste Zeilen, die String nicht enthalten *
 16:   * >output Ausgabeumlenkung                   *
 17:   * >>output Ausgabeumlenkung Append-Modus     *
 18:   *                                            *
 19:   * z.B. grep -i move.w *.s                    *
 20:   *      grep "int print (void)" *.c >prn:     *
 21:   *      grep -in include *.c *.s *.mac        *
 22:   *                                            *
 23:   * Rückgabe: 0 Zeile(n) gefunden              *
 24:   *           1 keine Zeile gefunden           *
 25:   *           2 Fehler aufgetreten             *
 26:   *                                            *
 27:   * 22.01.90 Jan Bolt                          *
 28:   *                                            *
 29:   * (c) MAXON Computer GmbH 1990               *
 30:   *============================================*/
 31:
 32:  #include <stdio.h>
 33:  #include <stdlib.h>
 34:  #include <ext.h>
 35:  #include <tos.h>
 36:  #include <string.h>
 37:
 38:  #define FALSE 0
 39:  #define TRUE 1
 40:  #define MAX_ARGS 30
 41:  #define MAX_FILES 10
 42:  #define MATCH 0
 43:
 44:  /* globale Variablen */
 45:  char myname[] = "grep";
 46:  unsigned long mlines = 0;  /* Anz gefundener
                                    Zeilen */
 47:
 48:  char *pattern = "";        /* Suchstring  */
 49:  /* Dateimuster */
 50:  char *files[MAX_FILES] = {NULL,NULL,NULL,NULL,
                              NULL,
 51:                           NULL,NULL,NULL,NULL,
                              NULL};
 52:  int cflag = FALSE;          /* c Option    */
 53:  int dflag = FALSE;          /* d Option    */
 54:  int hflag = FALSE;          /* h Option    */
 55:  int iflag = FALSE;          /* i Option    */
 56:  int lflag = FALSE;          /* l Option    */
 57:  int nflag = FALSE;          /* n Option    */
 58:  int vflag = FALSE;          /* v Option    */
 59:
 60:  void usage (void)
 61:       {
 62:       fprintf (stderr,"Usage: %s -cdhilnv pattern
                file ...\n",myname);
 63:
 64:       exit (2);
 65:       }
 66:
 67:  void error (int err,char *msg)
 68:       {
 69:       static char *errors[] = {"can't open",
 70:                                "invalid
                                     redirection to",
 71:                                "unterminated
                                     string"};
 72:
 73:       fprintf (stderr,"%s: %s %s\n",myname,
                errors[err-1],msg);
 74:
 75:       exit (2);
 76:       }
 77:
 78:  void message (char *msg)
 79:       {
 80:       fprintf (stderr,"%s: %s\n",myname,msg);
 81:       }
 82:
 83:  FILE *redirect (char *out,char *mode)
 84:       {
 85:       FILE *red;
 86:
 87:       if ((red=freopen(out,mode,stdout)) == NULL)
 88:          error (2,out);
 89:
 90:       return red;
 91:       }
 92:
 93:  /* Feststellen, ob pattern in zeile enthalten
         ist */
 94:  /* liefert Zeiger auf 1. Vorkommen von pattern
         in */
 95:  /* zeile oder NULL (erweiterungsfähig) */
 96:  char *get_match (char *zeile,char *pattern)
 97:       {
 98:       return strstr (zeile,pattern);
 99:       }
100:
101:  void do_file (char *file)
102:       {
103:       char zeile[256], uzeile[256];
104:       int match;
105:       unsigned long znr = 0;
106:
107:       while (fgets(zeile,255,stdin) != NULL)
108:             {
109:             strcpy (uzeile,zeile);
110:             znr++;
111:             if (iflag)
112:                strupr (uzeile);
113:             match = FALSE;
114:             if (get_match(uzeile,pattern) != NULL)
115:                match = TRUE;
116:             if (match ^ vflag)
117:                {
118:                mlines++;
119:                if (!cflag)
120:                   {
121:                   if (lflag)
122:                      {
123:                      printf ("%s\n",file);
124:                      break;
125:                      }
126:                   if (!hflag)
127:                      printf ("%s:",file);
128:                   if (nflag)
129:                      printf ("%lu:",znr);
130:                   printf ("%s",zeile);
131:                   }
132:                }
133:             }
134:       }
135:
136:  void do_grep (void)
137:       {
138:       int i, exist;
139:       int match = FALSE;
140:       DTA *dta_pnt;
141:       FILE *fp;
142:       char path[65];
143:       char *p;
144:
145:       dta_pnt = Fgetdta ();
146:       if (iflag)
147:          strupr (pattern);
148:       i = 0;
149:
150:       /* falls kein file angegeben, von stdin
             lesen */
151:       if (files[i] == NULL)
152:          {
153:          match = TRUE;
154:          do_file ("");
155:          return;
156:          }
157:
158:       while (files[i] != NULL)
159:             {
160:             /* Pfad isolieren */
161:             strncpy (path,files[i],64);
162:             path[64] = '\0';
163:             if ((p=strrchr(path,'\\')) != NULL)
164:                p++;
```

→

```
165:            else
166:               p = path;
167:
168:            exist = Fsfirst(files[i],0);
169:
170:            while (exist == MATCH)
171:                  {
172:                  match = TRUE;
173:                  strcpy (p,dta_pnt->filename);
174:                  strlwr (path);
175:                  if ((fp=freopen(path,"r",
                            stdin)) == NULL)
176:                     error (1,path);
177:
178:                  do_file (p);
179:                  fclose (fp);
180:                  exist = Fsnext ();
181:                  }
182:            i++;
183:            }
184:      if (!match)
185:         message ("no match.");
186:      else
187:      if (cflag)
188:         printf ("%lu\n",mlines);
189:      }
190:
191:  int do_main (int argc,char *argv[])
192:      {
193:      char opt;
194:      int i, k;
195:
196:      if (argc < 2)
197:         usage ();
198:
199:      /* Optionen lesen */
200:      i = 1;
201:      if (*(argv[i]) == '-')
202:         {
203:         for (k=1; (opt=*(argv[i]+k) & 0xdf) !=
                 '\0'; k++)
204:            {
205:            switch (opt)
206:                   {
207:                   case 'C' : cflag = TRUE; break;
208:                   case 'D' : dflag = TRUE; break;
209:                   case 'H' : hflag = TRUE; break;
210:                   case 'I' : iflag = TRUE; break;
211:                   case 'L' : lflag = TRUE; break;
212:                   case 'N' : nflag = TRUE; break;
213:                   case 'V' : vflag = TRUE; break;
214:                   }
215:            }
216:         i++;
217:         }
218:
219:      /* String lesen */
220:      if (argc <= i)
221:         usage ();
222:      pattern = argv[i++];
223:
224:      /* Dateimuster lesen */
225:      k = 0;
226:      while ((k < MAX_FILES) && (argc > i))
227:            files[k++] = argv[i++];
228:
229:      do_grep ();
230:
231:      if (dflag)
232:         {
233:         message ("hit Return to continue...");
234:         getch();
235:         }
236:
237:      if (mlines)
238:         return 0;
239:      else
240:         return 1;
241:      }
242:
243:  /*============== Hauptprogramm ==============*/
244:
245:  /* eigene Verarbeitung Kommandozeile, wegen " "
      und > */
246:
247:  int main ()
248:      {
249:      extern BASPAG *_BasPag;
                   /* Zeiger auf eigene Basepage */
250:      int argc, cmdlen;
251:      static char *argv[MAX_ARGS+1];
252:      char first;
253:      char *cmdline, *p, *a;
254:      static char cmd[128];
255:      static char rmode[] = "w";
256:
257:      cmdline = _BasPag->p_cmdlin;
                     /* Zeiger auf Kommandozeile */
258:      cmdlen = (int)*cmdline;
                     /* Länge Kommandozeile       */
259:      strncpy(cmd,&cmdline[1],cmdlen);
260:      cmd[cmdlen] = '\0';
261:
262:      argc = 1;
263:      argv[0] = myname;
264:
265:      p = cmd;
266:
267:      while ((argc <= MAX_ARGS) && (*p))
268:            {
269:            while ((*p) && (*p == ' '))
270:               p++;
271:            if (!*p)
272:               break;
273:            a = p;
274:            first = *p;
275:            switch (first)
276:                   {
277:                   case '\'' :
278:                   case '\"' :
279:                         a++;
280:                         while (*++p != first)
281:                            if (!*p)
282:                               error(3,"in
                                  commandline");
283:                         *p++ = '\0';
284:                         argv[argc++] = a;
285:                         break;
286:                   case '>'  :
287:                         if (*(++p) == '>')
288:                            {
289:                            rmode[0] = 'a';
290:                            p++;
291:                            }
292:                         while ((*p) && (*p == '
                                       '))
293:                            p++;
294:                         if (!*p)
295:                            error(2,"?");
296:                         a = p;
297:                         while ((*p) && (*p != '
                                       '))
298:                            p++;
299:                         if (*p == ' ')
300:                            *p++ = '\0';
301:                         redirect (a,rmode);
302:                         break;
303:                   default   :
304:                         while ((*p) && (*p != '
                                       '))
305:                            p++;
306:                         if (*p == ' ')
307:                            *p++ = '\0';
308:                         argv[argc++] = a;
309:                   }
310:
311:            }
312:
313:      return do_main(argc,argv);
314:      }
```

Listing: Grep sucht Strings in Dateien.

DIRECTORY ALS BAUMDIAGRAMM

Peter Ubachs

DAS DESKTOP STELLT MIT SEINEN FENSTERN ZWAR KOMFORTABEL AUSSCHNITTE DER ORDNERSTRUKTUR DAR, MÖCHTE MAN ABER EINEN ÜBERBLICK ÜBER DAS GESAMTE DIRECTORY EINER DISKETTE ODER GAR FESTPLATTEN-PARTITION GEWINNEN, REICHT DIE DARSTELLUNG NICHT MEHR AUS. EINES DER 'NORTON UTILITIES' (FÜR PCS) STELLT U.A. DIE ORDNERSTRUKTUR ALS BAUMDIAGRAMM DAR, GENAU DIES LEISTET AUCH DAS ABGEDRUCKTE GFA-BASIC-PROGRAMM.

Zunächst wird der Pfad ausgewählt, von wo aus der Dateibaum gezeichnet werden soll. Danach wird das Ausgabegerät bestimmt. Die Bildschirmausgabe läßt sich mit der linken Maustaste anhalten, mit der rechten geht es weiter. Bei der Datei-Ausgabe muß man noch den Namen der Ausgabedatei eingeben, dann wird der entsprechende Kanal geöffnet. Beim Ausdruck erfolgt keine Anpassung der Sonderzeichen (hier "|" und "\"). Diese muß man für seinen Drucker evtl. noch selber einfügen, oder man benutzt die residenten Anpassungsprogramme PTEPSON bzw. PTIBM, die dem GFA-BASIC beigelegt waren.

Wie wird nun das Baumdiagramm gezeichnet? Da Bäume rekursive Datenstrukturen darstellen, geht man beim Aufbauen und Zeichnen derselben am einfachsten auch rekursiv vor. Die Prozedur *direktory_zusammensuchen* ruft sich solange selber auf, bis keine weiteren Dateien mehr gefunden werden. Es sollte mit geringen Aufwand möglich sein, das Programm in andere Programmiersprachen, die Betriebssystem-Aufrufe und Rekursion zulassen, zu übertragen. Zur Dateisuche werden die GEMDOS-Funktionen *Fsfirst* und *Fsnext* verwendet. *Fsfirst* durchsucht das aktuelle Directory nach Dateien, auf die der angegebene Name paßt. Dabei bestimmt der zweite Parameter *"attribut"* nach welchen Dateienamen gesucht werden soll (16 bedeutet hierbei, daß auch nach Ordnern gesucht werden soll). Das Ergebnis wird in der DTA abgelegt. *Fsnext* setzt die mit *Fsfirst()* begonnene Suche fort. Nähere Informationen zu diesen Funktionen sind z.B. in [1] zu finden. Die Prozedur *dateinamen_bestimmen* liest den Dateinamen aus der DTA, vorsatz gibt die vor dem Dateinamen stehenden senkrechten Striche des Baumdiagrammes aus und schalt (gehörte ebenfalls zum Lieferumfang des GFA-BASICS) schaltet auf die 8x8-Kleinschrift um.



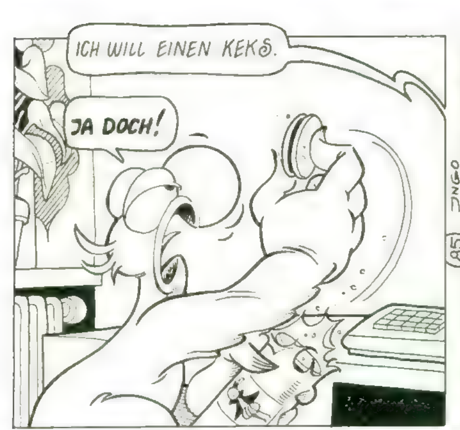
```
 1:  ' ++++++++++++++++++++++++++++++++++++++++++++++
 2:  ' DIRBAUM.BAS  Directory als einfaches
       Baumdiagramm darstellen / von Peter Ubachs
 3:  ' ++++++++++++++++++++++++++++++++++++++++++++++
 4:  ' + (c) MAXON Computer GmbH 1990
 5:  ' ++++++++++++++++++++++++++++++++++++++++++++++
 6:  '
 7:  ' benutzte GEMDOS-Funktionen
 8:  DEFFN fsfirst(adr_dateiname%,attribut%)=
           GEMDOS(&H4E,L:adr_dateiname%,attribut%)
 9:  DEFFN fsnext=GEMDOS(&H4F)
10:  @schalt(4) !8x8 Kleinschrift
11:  '
12:  max_dateien%=2500
13:  max_ebenen%=15
14:  DIM name$(max_dateien%,max_ebenen%)
15:  DIM anz_dateien%(max_ebenen%),nr%(max_ebenen%)
16:  '
17:  einrueck%=10
```

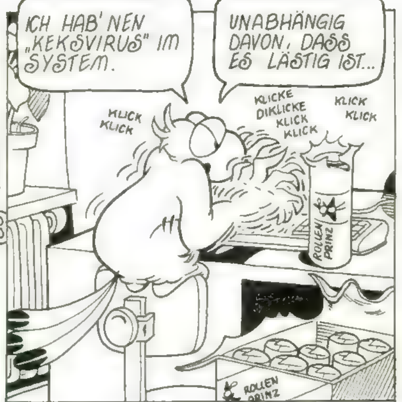
```
18:  '
19:  DO
20:    ' Welcher Pfad?
21:    FILESELECT DIR$(0)+"\*.*","",pfad$
22:    ' --------------------------------------------
23:    EXIT IF pfad$="" !~Abbruch angeklickt?
24:    ' --------------------------------------------
25:    WHILE RIGHT$(pfad$)<>"\"
26:      pfad$=LEFT$(pfad$,LEN(pfad$)-1)
27:    WEND
28:    CHDIR pfad$
29:    '
30:    ' Ausgabegerät?
31:    ALERT 2,"A u s g a b e g e r ä t ?",1,
               "Monitor|Drucker|Datei",antw%
32:    IF antw%=1
33:      OPEN "O",#1,"CON:"
34:      CLS
35:    ELSE
```

→

```
36:       IF antw%=2
37:         OPEN "O",#1,"PRN:"
38:       ELSE
39:         FILESELECT DIR$(0)+"\*.*","",dirpfad$
40:         OPEN "O",#1,dirpfad$
41:       ENDIF
42:     ENDIF
43:     '
44:     VOID GEMDOS(&H1A,L:BASEPAGE+128) !DTA setzen
45:     PRINT #1;pfad$
46:     ebene%=1
47:     @directory_zusammensuchen(ebene%,pfad$)
48:     '
49:     IF antw%=1
50:       PRINT "                                        "+
              CHR$(27)+"p Taste drücken "+CHR$(27)+"q";
51:       REPEAT
52:       UNTIL LEN(INKEY$)
53:     ENDIF
54:     CLOSE
55:   LOOP
56:   @schalt(8)
57:   EDIT
58:   '
59:   PROCEDURE directory_zusammensuchen(ebene%,pfad$)
60:     LOCAL fsret%,dateiname$,datei_nr%,suchpfad$
61:     suchpfad$=pfad$+"*.*"+CHR$(0)
62:     datei_nr%=0
63:     fsret%=FN fsfirst(VARPTR(suchpfad$),16)
64:     WHILE fsret%=0
65:       @dateiname_bestimmen(*dateiname$) !d.h. aus
                                            der DTA lesen
66:       IF dateiname$<>"." AND dateiname$<>".."
67:         INC datei_nr%
68:         name$(datei_nr%,ebene%)=dateiname$
                                              !abspeichern
69:       ENDIF
70:       fsret%=FN fsnext
71:     WEND
72:     anz_dateien%(ebene%)=datei_nr%
73:     '
74:     '
75:     IF anz_dateien%(ebene%)>0
76:       @vorsatz(ebene%)
77:       PRINT #1;"|"
78:     ENDIF
79:     '
80:     nr%(ebene%)=1
81:     WHILE nr%(ebene%)<=anz_dateien%(ebene%)
82:       IF MOUSEK=1 !linke Maustaste: Halt
83:         REPEAT
84:         UNTIL MOUSEK=2 !rechte Maustaste: Weiter
85:       ENDIF
86:       @vorsatz(ebene%)
87:       PRINT #1;"|_____ ";name$(nr%(ebene%),ebene%)
88:       @directory_zusammensuchen(ebene%+1,pfad$+
                    name$(nr%(ebene%),ebene%)+"\")
89:       INC nr%(ebene%)
90:     WEND
91:     '
92:     IF (anz_dateien%(ebene%)>0) AND (nr%(ebene%-
          1)<anz_dateien%(ebene%-1))
93:       @vorsatz(ebene%)
94:       PRINT #1
95:     ENDIF
96:     '
97:   RETURN
98:   '
99:   PROCEDURE dateiname_bestimmen(adr_name%)
100:    LOCAL name$,i%
101:    i%=BASEPAGE+128+30
102:    WHILE PEEK(i%)
103:      name$=name$+CHR$(PEEK(i%))
104:      INC i%
105:    WEND
106:    *adr_name%=name$
107:  RETURN
108:  '
109:  PROCEDURE vorsatz(ebene%)
110:    LOCAL l%
111:    IF ebene%>1
112:      FOR l%=1 TO ebene%-1
113:        IF nr%(l%)<anz_dateien%(l%)
114:          PRINT #1;"|";SPACE$(einrueck%-1);
115:        ELSE
116:          PRINT #1;SPACE$(einrueck%);
117:        ENDIF
118:      NEXT l%
119:    ENDIF
120:  RETURN
121:  '
122:  PROCEDURE schalt(x)
123:    a$=MKI$(&HA000)+MKI$(&H2009)+MKI$(&H4E75)
124:    a=VARPTR(a$)
125:    a=C:a()
126:    LPOKE INTIN,LPEEK(a+x)
127:    DPOKE CONTRL+2,0
128:    DPOKE CONTRL+6,0
129:    DPOKE CONTRL+10,102
130:    VDISYS 5
131:  RETURN
132:  ' @schalt(4)=Kleinschrift 8*8
133:  ' @schalt(8)=Grossschrift 8*16
```

ROCKUS









for better communication

BEST 1200 PLUS *	300,1200 Bit/s	198,-
BEST 2400 L *	300,1200,2400 Bit/s	328,-
BEST 2400 PLUS *	300,1200,1200/75,2400 Bit/s	438,-
BEST 2400 EC MNP 5 *	300,1200,2400 Bit/s	548,-
SUPRA 2400 zi	Steckkarte für Amiga 2000	348,-

Autorisierter Distributor • Auf BEST-Modems 12 Monate Garantie
* Diese Modems mit deutschem Handbuch

Der Anschluß der Modems am Netz der Deutschen Bundespost Telekom ist strafbar !



Btx/Vtx mit dem *ATARI*

MultiTerm pro

Wir setzen neue Maßstäbe !

Btx/Vtx-Darstellung mit Graustufen und 32/4096 Farben
Mit Automatischem Makro Generator AMG
und Programmiersprache MPL

An Modem/Akustikkoppler 158,- • An D-BT03 236,-

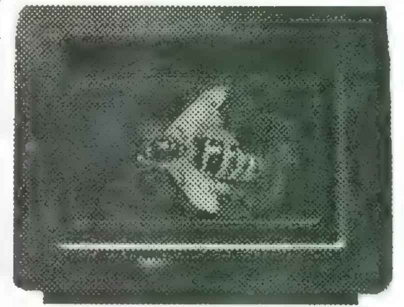
Schweiz: *tribatech ag* Unterführungsstr. 29, 4601 Olten, Tel: 062- 260222

Fordern Sie Informationsmaterial an !
Händleranfragen erwünscht !



Projensdorfer Str. 14 • 2300 Kiel 1
Tel: 0431 - 33 78 81 • Fax: 0431 - 3 59 84
Btx: * TKR #

SM 124 Multisync II



1 Monitor
3 Auflösungen
100 % Softwarekompatibel
Hilfsprogramme auf Disk
Beste Industriequalität
Als Bausatz lieferbar

Nach der fachgerechten Umrüstung ist Ihr Monitor SM 124 in der Lage, in allen 3 Auflösungen zu arbeiten. Die Farben werden dabei in Graustufen dargestellt. Die hohe Auflösung verliert nichts an ihrer Brillanz. Die Softwarekompatibilität wird durch diese reine Hardwarelösung nicht beeinträchtigt.

Werten Sie Ihren SM 124 auf !

Umbau Ihres Monitors (ca. 3 Tage)	DM 249,00
Bausatz komplett bestückt*	DM 149,00
Leerplatine incl. aller Bauteile*	DM 129,00

*ausführliche Anleitung (dt.) liegt bei.

Bei Bestellung bitte Baujahr des Monitors angeben.

Preise zzgl. Porto und Verpackung DM 5,00, NN plus DM 2,00, besser V-Scheck (**Ausland nur Euro-Scheck**).

Händleranfragen erwünscht !

iks
In der Au 22
7516 Karlsbad 4
07202/7687

Hendrik Haase Computersysteme
präsentiert:

Atari-Computer

Atari 1040 STE	Preissenkung
Atari Mega ST 1	Preissenkung
Atari Mega ST 2	Preissenkung
Atari Mega ST 4	Preissenkung
Atari Megafile 30 Festplatte	850,- DM
Atari Megafile 60 Festplatte	1240,- DM
Atari Megafile 44 Wechselplatte	Preissenkung
Vortex Datajet 30 Festplatte	1100,- DM
Vortex Datajet 60 Festplatte	1690,- DM
Star Drucker LC 24/10	668,- DM
Epson LQ 550	698,- DM
HP Deskjet	1050,- DM
HP Deskjet plus	1690,- DM
Atari Laserdrucker SLM 804	2600,- DM
Graustufen Multisync	550,- DM
Mitsubishi Multisync	1298,- DM

Stacy - Computer
in unterschiedlichen Versionen

Bestellungen und Informationen bei:

Hendrik Haase Computersysteme
Wiedfeldtstraße 77 • D-4300 Essen 1
Telefon 02 01 - 42 25 75 • Fax 02 01 - 41 04 21

UMSON ST ...

bekommen Sie zwar außer unserer Info nichts ... bei uns, jedoch aktuelle und leistungsfähige Programme zu fairen Preisen !

COMPTABLE ST (Buchführung) DM 198.00
Ein Buchführungsprogramm für Geschäft und Privat. 500 Konten (DATEV). 10 Steuersätze, bis zu 10000 Buchungen. Steuersätze und Privatanteilsätze können bereits im Kontenrahmen vorgegeben werden. Auswertungen: Saldenliste, Kontenblätter, Journal, Kassenbuch, Gewinn/Verlust-Rechnung, Umsatzsteuerdaten jeweils für einen beliebigen Monat, ein beliebiges Quartal oder das ganze Jahr! Alle Ausgaben auf Bildschirm, Drucker oder Datei. Buchungsmemo für 25 Buchungen (einfach durch Anklicken abrufbar). Universelle Druckeranpassung. Incl. 50- seitigem, reichhaltig illustriertem Handbuch (mit Ausdruckbeispielen) im Ringordner!

K-FAKT ST (Professionelles Fakturierungssystem) DM 398.00
Adress-, Artikel und Stücklistenverwaltung, Umsatz/Absatzstatistik, frei wählbare Indexfelder. Erstellung von Angebot, Auftr.-Best., Lieferschein, Rechnung, Storno, Gutschrift, Mahnung, Anfrage, Bestellung, Eingangsbeleg, Aufmaßberechnungen. Alle Arten von Listen und Formularen. Anbindung an 1st Word Plus. Besteuerung nach Artikel oder Kunde, somit auch für Steuersystem in der Schweiz geeignet! Universeller Druckertreiber, an alle Drucker anzupassen. Incl. ausführlichem, über 100 Seiten starken Handbuch mit zahlreichen Illustrationen. DEMO-VERSION DM 20,-- HANDBUCH VORAB DM 50,- (wird angerechnet !)

ST-MATHETRAINER II (Lernprogramm) DM 59.00
Für 1-6 Schuljahr 1x1 und Mischrechnen mit wählbaren Höchstzahlenwerten. Umrechnung von Gewichten und Längenmaßen, im Schwierigkeitsgrad durch eingebaute Editierfunktion frei an den Lernbedarf bzw. den Schulbuchstoff anzupassen. Mit Benotung und Protokoll. Voll in GEM eingebunden! Incl. deutscher Bedienungsanleitung

ST-RECHTSCHREIBEN II (Lernprogramm) DM 59.00
Für 1-6 Schuljahr. Wörter in Sätze einfügen, Singular und Plural, Kommata setzen. Im Schwierigkeitsgrad durch eingebaute Editierfunktion frei an den Lernbedarf bzw. Schulbuchstoff anzupassen. Mit Benotung und Protokoll. Voll in GEM eingebunden! Incl. deutscher Bedienungsanleitung.

TKC-VIDEO (Verwaltung von Videocassetten) DM 79.00
Verwaltet bis zu 5000 Videofilme pro Datei. 10 Felder für Titel, Spielzeit, Zählwerk, Darsteller, "verliehen an", Bemerkungen, usw. Umfangreiche Such- u. Selektierfunktionen. Ausdruck von Listen + Etiketten (frei über ASCII-Maske zu gestalten). Voll unter GEM, deutsche Bedienungsanleitung.

TKC-MUSICBOX (Musiktitelverwaltung) DM 79.00
Verwaltet bis zu 5000 CDs, LPs oder MCs. Suche nach Einzeltiteln, Audruck von Haupttitelübersicht u. Gesamttitelliste mit MC-Nummern und Etiketten. Voll unter GEM, incl. deutscher Bedienungsanleitung.

TKC-TRAINER (Trainingsprogramm) DM 99.00
Sehr hoher Lernerfolg durch Karteikastenprinzip. Für Sprachen, Chemie, Physik, etc. Voll unter GEM, leicht auch für Schüler zu bedienen. Incl. deutschem Handbuch

TKC-BANKMANAGER (mit Sammler-Ausdruck) DM 99.00

ST-GIRO PLUS DM 49.00
Überweisungsdruckprogramme für den geschäftlichen (häufigen) und privaten Gebrauch. Auch für Schecks + Lastschriften. Ausdruck über ASCII-Masken an alle Formulare anzupassen!

ST-VOKABELTRAINER (Trainingsprogramm) DM 49.00
Vokabeltraining für Englisch, Französisch, Italienisch, Spanisch und zeichenkompatible Sprachen. 100 Vokabeln pro Datei, voll unter GEM

TKC-TERMIN/ADRESS (Terminkalender) DM 99.00
Universeller Terminplaner mit integrierter Adressverwaltung. Erstellung von Terminlisten, Adresslisten und Serienbriefen. Voll unter GEM, incl. ausführlicher Bedienungsanleitung!

TK COMPUTER-TECHNIK Thomas Kaschadt
Hard- & Software Entwicklung & Vertrieb

Bischofsheimer Straße 17 * D-6097 Trebur-Astheim, Fernruf: (06147) 3550
Postfach 60, FAX 06147-3555, Btx. 06147-3555
24-Stunden-Auftragsannahme per Anrufbeantworter !

MAKE_OBJECT

LINKFÄHIGE OBJEKTDATEIEN LEICHTGEMACHT

Ralf Plamitzer

WER HAT NICHT SCHON EINMAL EIN PROGRAMM GESCHRIEBEN, DAS GEWISSE HILFSDATEIEN BENÖTIGT? SOLLEN Z.B. BILDER ANGEZEIGT WERDEN, MÜSSEN SIE ERST MÜHSELIG VON DISKETTE NACHGELADEN WERDEN.

MAKE_OBJ.TTP schafft hier Abhilfe. Es macht aus normalen Dateien linkfähige Objektdateien im weit verbreiteten DR-Format. Somit können beliebige Daten direkt ins Programm integriert werden. Dies können z.B. Bilder, Zeichensätze, Ressourcen oder ähnliches sein. Der Phantasie sind keine Grenzen gesetzt, solange man den Aufbau der Dateien kennt. Die erzeugten Objektdateien müssen dann nur noch zum Hauptprogramm hinzugelinkt werden.

Dem Programm wird eine Kommandozeile übergeben, die folgendes Format hat: *PFAD SYMBOLNAME*. PFAD ist der komplette Name der zu bearbeitenden Datei, z.B. B:\ORDNER\NAME.EXT. SYMBOLNAME ist ein maximal 8 Zeichen langer Name, der in die Symboltabelle eingetragen wird. Achtung: Groß-/Kleinschreibung beachten! MAKE_OBJ.TTP erzeugt dann eine Datei namens B:\ORDNER\NAME.O. Das Anhängen der Extension 'O' erledigt übrigens die Funktion set_ext(..). Als erstes wird überprüft, ob die Kommandozeile dem erwarteten Format entspricht. Danach wird die zu bearbeitende Datei geöffnet, deren Größe ermittelt und entsprechend viel Speicher reserviert. Die Daten werden eingelesen und die Datei geschlossen, danach eine neue Datei erzeugt, und der Objekt-Header, die Daten und die Symboltabelle geschrieben. Objekt-Header und Symboltabelle sind schon vordefiniert, da sich die Daten bis auf den Symbolnamen und die Länge des Databereichs nicht ändern.

Daran schließen sich noch eine ganze Menge Nullen an. Die Datei wird geschlossen und der reservierte Speicher wieder freigegeben. Um die Arbeitsweise von MAKE_OBJ.TTP zu verdeutlichen, nun zu einem kleinen Beispiel. DEMO1.C zeigt, wie man ein im Programm integriertes Bild (im Screen-Format) anzeigen kann. Natürlich muß die Bilddatei vorher mit MAKE_OBJ.TTP bearbeitet werden. Man übergibt dem Programm den kompletten Pfad der Datei (z. B. "B:\MY_PIC.DOO") und den gewünschten Symbolnamen (in unserem Fall "BILD"). MAKE_OBJ.TTP erzeugt dann eine Objektdatei mit dem Namen "B:\MY_PIC.O". Diese Datei wird später zum Hauptprogramm "hinzugelinkt".

Das Listing

Zum Listing ist eigentlich nicht viel zu sagen. Als erstes deklarieren Sie "BILD" als "extern long". Man muß jetzt nur noch die Adresse des Bildschirmes ermitteln, und pic die Adresse des Bildes im Speicher zuweisen. Danach lassen sich die Bilddaten direkt auf den Bildschirm kopieren. Will man Resource-Dateien ins Programm einbinden, gestaltet sich die Sache schon etwas komplizierter. Im DEMO2.C werden zwei Ressourcen gleichzeitig angesprochen.

Man muß dabei nichts anderes tun, als die Funktion rsrc_load(..), die man im Normalfall benutzen würde, durch rsrc_init(..) zu ersetzen. Übrigens kann dann rsrc_free() wegfallen. Die Adressen der Objektbäume, freien Strings etc. können wie üblich über die Funktion rsrc_gaddr(..) ermittelt werden. Bestehende Programme muß man also nur geringfügig ändern. Als Demonstration legen Sie zwei Resourcefiles an ("RSCA.RSC" und "RSCB.RSC"), wobei jeweils ein Objektbaum, dessen Aufbau beliebig ist, erstellt wird. Den ersten nennen wir TREE1, den zweiten TREE2. Verwendet man wie hier mehrere Resource-Dateien, muß man natürlich darauf achten, daß man nicht zweimal den gleichen Namen an verschiedene Objekte vergibt.

Beide Dateien werden nun mit "MAKE_OBJ.TTP" bearbeitet. Aufruf: 'RSCA.RSC RSCA' bzw. 'RSCB.RSC RSCB' Die Hauptarbeit erledigt also die Funktion rsrc_init(..). Ihr übergibt man die Anfangsadresse der zu bearbeitenden Resource-Datei (rsrc_init(&RSCA)). Rsh zeigt dann automatisch auf den Resource-Header.

Nun wird in global[9] die Größe der Ressource, die in rsh_rssize steht, geschrieben, danach in global[7] und global[8] die Adresse der Datei, und in global[5] und global[6] die Adresse einer Tabelle mit den Anfangsadressen der Objektbäume. Der Anfang dieser Tabelle steht wiederum im Header (rsh_trindex). In rsh_trindex steht der Beginn dieser Tabelle relativ zum Anfang der Resource-Datei. Man muß also nur noch 'rsh hinzuaddieren' [das erledigt das Makro add_ptr(..)] und ptr zuweisen. Diese Zeilen sind notwendig, damit später die Funktion rsrc_gaddr(..) richtig

arbeiten kann. Ptr zeigt also jetzt auf den ersten Eintrag der Tabelle. Hier könnte z.B. eine 36 stehen, was bedeuten würde, daß der erste Objektbaum ab Byte Nummer 36 in der Resource-Datei zu finden ist. Wir addieren wieder 'rsh', um die wirkliche Adresse zu erhalten. Dies geschieht innerhalb einer Schleife, in der alle Bäume durchlaufen werden. Analoges passiert mit den freien Strings und den freien Images. Anschließend werden für alle Objekte nach dem gleichen Prinzip mit dem Makro add_rsh(..) die Zeiger z.B. auf Texte etc. gesetzt und die Koordinaten mittels rsrc _obfix(..) angepaßt.



```
 1:  /*********************************\
 2:  |  Von   Ralf Plamitzer           |
 3:  |        Dresdenerstr. 7          |
 4:  |        6120 Michelstadt         |
 5:  |        TURBO C 2.0              |
 6:  |                                 |
 7:  |  (c) MAXON Computer GmbH 1990   |
 8:  |  Macht aus beliebigen Dateien   |
 9:  |  linkbare Objektdateien im      |
10:  |  DR-Format.                     |
11:  |  Beispielaufruf:                |
12:  |  B:\ORDNER\FILE.EXT  SYMBOL     |
13:  |  erzeugt:  B:\ORDNER\FILE.O     |
14:  \*********************************/
15:
16:  #include <stdlib.h>
17:  #include <stdio.h>
18:  #include <string.h>
19:  #include <ext.h>
20:
21:  typedef struct
22:  {
23:   int               branch; /*  Immer 0x601A      */
24:   long    tlen;         /*  Länge Textsegment (0) */
25:   long    dlen;         /*  Länge Datensegment    */
26:   long    blen;         /*  Länge BSS             */
27:   long    slen;         /*  Länge Symb.tab. (14)  */
28:   char    reserved[10];
29:  }OHEADER;
30:
31:  typedef struct
32:  {
33:   char    name[8];      /*  Symbolname */
34:   int               typ; /*  Bei uns immer 0xA400 */
35:   long    fill;         /*  Immer null */
36:  }SYM;
37:
38:  OHEADER  oh = {
          0x601A,0L,0L,0L,0x0EL,0,0,0,0,0,0,
                      0,0,0,0 };
39:  SYM      sym = { "12345678",0xA400,0L };
40:
41:
42:  void set_ext(char *name,char *ext)
43:  {
44:   char *c;
45:
46:   if((c = strrchr(name,'.')) == NULL)
47:          return;
48:   *++c = '\0';
49:   strcat(name,ext);
50:  }
51:
52:  int main(int argc,char *argv[])
53:  {
54:   FILE *file;
55:   char *buf,path[128];
56:
57:   if(argc != 3)
58:          return(1);
59:   strcpy(path,argv[1]);
60:   if((file = fopen(path,"rb")) == NULL)
61:          return(1);
62:   if((oh.dlen = filelength(file->Handle)) % 2)
63:          oh.dlen++;
64:   if((buf = malloc(oh.dlen)) == NULL)
65:   {
66:          fclose(file);
67:          return(1);
68:   }
69:   fread(buf,oh.dlen,1,file);
70:   fclose(file);
71:   set_ext(path,"O");
72:   if((file = fopen(path,"wb")) == NULL)
73:   {
74:          free(buf);
75:          return(1);
76:   }
77:   strncpy(sym.name,argv[2],8);
78:   fwrite(&oh,sizeof(OHEADER),1,file);
79:   fwrite(buf,oh.dlen,1,file);
80:   fwrite(&sym,sizeof(SYM),1,file);
81:   memset(buf,0,oh.dlen);
82:   fwrite(buf,oh.dlen,1,file);
83:   fclose(file);
84:   free(buf);
85:   return(0);
86:  }
```

Listing: So werden in das DR-Format gewandelt.

DIE MAUS IN RENTE

Buttons über Tastatur

Uwe Seimet

Ohne Zweifel gestaltet sich die Bedienung der Dialogboxen beim GEM durch Verwendung der Maus recht komfortabel. Aber nicht jeder Anwender möchte innerhalb von Dialogen ständig zur Maus greifen, um Buttons selektieren oder deselektieren zu können.

Insbesondere innerhalb von Programmen, die in der Hauptsache tastaturorientiert arbeiten (z.B. bei Textverarbeitungen), ist es umständlich, ständig zur Maus greifen zu müssen, um Dialoge zu führen. Viele Programme ermöglichen zwar das Aufrufen von Menüoptionen über die Tastatur, aber was bringt es, wenn man dann doch wieder zur Maus greifen muß, um einen sich anschließenden Dialog zu führen?

Bereits in [1] wurde eine Möglichkeit vorgestellt, Buttons innerhalb von Dialogen über alternative Tastenkombinationen zu bedienen. Dabei stellt sich jedoch noch das Problem, daß man den Knöpfen natürlich nicht ansehen kann, welche Tastenkombination für welchen Knopf zuständig ist. Hier kann aber Abhilfe geschaffen werden. Das Verfahren, das ich hierzu vorstellen möchte, ist sehr komfortabel und macht intensiven Gebrauch von ansonsten recht selten verwendeten Möglichkeiten des GEM. Für meine Ausführungen setze ich voraus, daß Sie bereits eigene Dialoge mit einem Resource Construction Set erstellt und mit Hilfe der *form_do*-Routine verwaltet haben. Obwohl das zugehörige Beispielprogramm in Assembler geschrieben ist, sind Assembler-Kenntnisse nicht unbedingt notwendig, da alle Systemaufrufe in ähnlicher Form auch von C oder Pascal aus getätigt werden können.

Zunächst soll aber einmal die Zielsetzung umrissen werden:

- Buttons in Dialogen sollen mit alternativen Tastenkombinationen über die Tastatur bedienbar sein.
- Welche Taste einem Button zugeordnet ist, soll direkt aus der Dialogbox hervorgehen.
- Das Verfahren soll möglichst flexibel sein.
- Natürlich sollen ausschließlich dokumentierte Eigenschaften des GEM verwendet werden.

Am Ende unserer Arbeit soll der in Abbildung 1 dargestellte kleine Beispieldialog stehen, bei dem der Knopf "Radiobutton 1" über [Alternate][1], "Radiobutton 2" über [Alternate][2] und der "Abbruch"-Button über die [Undo]-Taste bedient werden kann.

Anhand von Abbildung 1 ist bereits gut zu erkennen, wie Punkt 2 unserer Vorgaben erfüllt werden soll: Der Buchstabe auf der Tastatur, der in Verbindung mit der Alternate-Taste gedrückt werden muß, um einen Button anzusprechen, wird innerhalb des Button-Textes unterstrichen dargestellt. An dieser Stelle möchte ich nun ansetzen, um den Aufbau einer eigenen Dialogroutine zu erläutern.

Die Objektstruktur des GEM

Alle Bestandteile einer Dialogbox stellen GEM-Objekte dar, deren Aussehen durch die Datenstruktur *OBJECT* vorgegeben wird: (s.Listing 1)

Ein besonders wichtiges Element dieser Struktur ist der Objekttyp, der eine grundlegende Aussage darüber macht, um welche Art von Objekt es sich überhaupt handelt.

Das GEM stellt eine Reihe von Standardobjekttypen zur Verfügung, mit denen die üblichen Dialoge aufgebaut werden können. Die Radiobuttons in Bild 1 entsprechen im Prinzip dem Objekttyp *G_BUTTON* - wenn da nicht jeweils der unterstrichene Buchstabe im Text vorhanden wäre. Diesen Texteffekt innerhalb der Buttons unterzubringen, ist nicht ohne weiteres möglich. Der Pointer *ob_spec* zeigt bei Buttons stets auf einen String mit dem Text des Objekts. Es gibt jedoch keine Möglichkeit, in den Daten für den Typ *G_BUTTON* eine Information darüber abzulegen, daß ein Buchstabe des Textes innerhalb des Dialogs unterstrichen, also mit einem besonderen Textattribut dargestellt werden soll. Da das GEM standardmäßig keinen Objekttyp zur Verfügung stellt, der es uns ermöglicht, Radiobuttons gemäß Abbildung 1 zu erzeugen, müssen wir also in die Trickkiste greifen und einen solchen Typ selbst definieren

Benutzer-definierte Objekte

Der Objekttyp *G_USERDEF* ermöglicht es, während des Aufbaus einer Dialogbox eigene Objekttypen darzustellen. *ob_spec* zeigt bei Objekten des Typs *G_USERDEF* auf eine *USERBLK*-Struktur: (s. Listing 2)

ub_code stellt einen Pointer dar, der auf eine Routine zeigt, die vom AES immer dann aufgerufen wird, wenn ein Objekt neu gezeichnet werden muß. Dieser Routine wird ein Zeiger auf eine *PARMBLK*-Struktur übergeben, die das zu zeichnende Objekt näher beschreibt: (s.Listing 3)

PARMBLK stellt der Routine, die über *ub_code* definiert wird, alle Angaben zur Verfügung, die benötigt werden, um das Zeichnen eines Objekts selber durchzuführen. Wie man sieht, hat man sich durch einige Strukturen durchzukämpfen, bevor es richtig zur Sache geht.

Eine kurze Denkpause

Wenn wir an dieser Stelle unser bisheriges Wissen rekapitulieren, sieht es so aus, daß zunächst statt des Objekttyps *G_BUTTON* der Typ *G_USERDEF* für alle Buttons verwendet werden muß, die von einer eigenen Zeichenroutine dargestellt werden sollen. *ob_spec* zeigt bei diesen Objekten nicht auf den Text der Buttons, sondern auf eine *USERBLK*-Struktur, die wiederum einen Zeiger auf unsere Zeichenroutine *ub_code* enthält. Diese Routine wird vom AES zu einem gegebenen Zeitpunkt aufgerufen und hat die Aufgabe, zunächst den Rahmen des Buttons und dann den Text innerhalb des Rahmens zu zeichnen. Hierbei muß ein Zeichen unterstrichen werden. Außerdem wird *ub_code* immer dann vom GEM aufgerufen, wenn der Status des Buttons geändert wird.

Wo ist der Haken?

Prinzipiell dürfte das beschriebene Vorgehen einsichtig sein, aber leider gibt es noch ein paar Probleme.

Radiobutton 1 Radiobutton 2
OK ABBRUCH

Abb. 1: Beispieldialog

```
typedef struct
{
        int          ob_next;    /* nächstes Objekt */
        int          ob_head;    /* erstes Kind */
        unsigned int ob_tail;    /* letztes Kind */
        unsigned int ob_type;    /* Objekttyp */
        unsigned int ob_flags;   /* Objektflags */
        unsigned int ob_state;   /* Objektstatus */
        char         *ob_spec;   /* Zeiger auf weitere Struktur */
        int          ob_x;       /* x-Position, relativ zum Parent */
        int          ob_y;       /* relative y-Position */
        int          ob_width;   /* Breite */
        int          ob_height;  /* Höhe */
} OBJECT;
```

Listing 1

```
typedef struct
{
        int  (*ub_code)();  /* Zeiger auf eigene Zeichenfunktion */
        long ub_parm;       /* optionaler Parameter */
} USERBLK;
```

Listing 2

```
typedef struct
{
        OBJECT *pb_tree;        /* Zeiger auf Objektbaum */
        int    pb_obj;          /* Objektnummer */
        int    pr_prevstate;    /* vorheriger Objektstatus */
        int    pr_currstate;    /* neuer Objektstatus */
        int    pb_x;            /* x-Position des Objekts */
        int    pb_y;            /* y-Position des Objekts */
        int    pb_w;            /* Breite des Objekts */
        int    pb_h;            /* Höhe des Objekts */
        int    pb_xc;           /* x-Pos. des Begrenzungsrechtecks */
        int    pb_yc;           /* y-Pos. des Begrenzungsrechtecks */
        int    pb_wc;           /* Breite des Begrenzungsrechtecks */
        int    pb_hc;           /* Hohe des Begrenzungsrechtecks */
        long   pb_parm;         /* Parameter aus USERBLK-Struktur */
} PARMBLK;
```

Listing 3

Beim Erstellen einer Dialogbox über ein Resource Construction Set gibt es keine Möglichkeit, den Objekttyp *G_USERDEF* zu vergeben. Es ist nur möglich, Objekte zu erstellen, die den Standardtypen entsprechen. Dies ist auch nicht weiter verwunderlich, denn das RCS kann ja nicht wissen, was sich hinter einem *G_USERDEF*-Objekt verbergen soll, da die Routine zum Zeichnen eines solchen Objekts erst im eigentlichen Programm vorhanden ist.

So bleibt zunächst nur die Möglichkeit, im RCS einen "normalen" Button als Objekt zu definieren und den Objekttyp später programmgesteuert von *G_BUTTON* in *G_USERDEF* zu ändern. Nun soll aber nicht unbedingt jeder Button in ein benutzerdefiniertes Objekt umgewandelt werden. Es wäre also ungünstig, im eigenen Programm grundsätzlich jedes Objekt des Typs *G_BUTTON* in ein *G_USERDEF*-Objekt zu verwandeln. Dies soll ja nur bei Buttons geschehen, die später ein besonderes Aussehen erhalten sollen. Wir müssen deshalb eine Möglichkeit finden, innerhalb des RCS festzulegen, welcher Button später in ein *G_USERDEF*-Objekt umdefiniert werden muß.

Verwendung eigener Objekt-Flags

Eine entsprechende Information kann in den Objekt-Flags der OBJECT-Struktur abgelegt werden. *ob_flags* enthält einige Bits, die vom GEM nicht verwendet werden und dem Programmierer für eigene Anwendungen zur Verfügung stehen. Somit wollen wir uns in Bit 10 der Objekt-Flags merken, ob ein Objekt im Programmverlauf in ein *G_USERDEF*-Objekt überführt werden soll. Bei Buttons, die über unsere eigene Zeichenroutine dargestellt werden sollen, wird dieses Bit gesetzt, ansonsten bleibt es gelöscht. (Die bekannten Resource Construction Sets für den Atari ST bieten alle die Möglichkeit, die nicht vom GEM reservierten Flags einzeln zu setzen.) Somit hätten wir das Problem, im RCS nicht direkt ein *G_USERDEF*-Objekt erzeugen zu können, erschlagen. Aber noch sind wir nicht am Ziel angelangt.

Wo bleibt die Textinformation?

Aufmerksame Leser werden vielleicht schon erkannt haben, wo die nächste Schwierigkeit liegt: *ob_spec* zeigt bei Objekten des Typs *G_BUTTON* auf den Text des Buttons, bei *G_USERDEF*-Objekten enthält *ob_spec* jedoch einen Zeiger auf eine *USERBLK*-Struktur. Dummerweise benötigen wir beide Zeiger, um mit unserer haus-

gemachten Zeichenroutine den Text des Buttons ausgeben zu können. Es ist also nicht möglich, nach dem Programmstart in die *ob_spec*-Pointer der Buttons einfach die Adresse der *USERBLK*-Struktur einzutragen, da durch dieses Vorgehen die Textinformation unwiderruflich verlorengehen würde. Der alte Inhalt von *ob_spec* muß demnach anderweitig untergebracht werden. Hierfür bietet sich der Parameter *ub_parm* innerhalb der *USERBLK*-Struktur an, der für den Programmierer reserviert ist.

Da für jedes benutzerdefinierte Objekt eine solche Struktur zur Verfügung gestellt werden muß, kann der Speicherplatzbedarf bei umfangreichen Ressourcen unangenehm groß werden. Schließlich benötigt jede *USERBLK*-Struktur 8 Bytes, so daß der Platzbedarf bei größeren Projekten bereits im Kilobyte-Bereich liegen kann.

Es ist deshalb nicht unbedingt sinnvoll, gleich nach dem Programmstart für alle *G_USERDEF*-Objekte eine *USERBLK*-Struktur zu erzeugen. Um Speicherplatz zu sparen, empfiehlt es sich, dies nur für den jeweils nächsten Dialog durchzuführen. Im Klartext heißt dies: Vor dem Zeichnen einer Dialogbox werden die *G_BUTTON*-Objekte in Objekte des Typs *G_USERDEF* überführt und die benötigen Datenstrukturen erzeugt. Nach Verlassen des Dialogs werden diese Änderungen wieder rückgängig gemacht. Durch dieses Verfahren befinden sich so wenig *USERBLK*-Strukturen wie möglich im Speicher.

Wir kommen der Lösung näher

Fassen wir an dieser Stelle erneut zusammen: Die Verwaltung von Objekten des Typs *G_USERDEF* muß bereits beim Erstellen des Dialogs im RCS vorbereitet werden. Hierzu markieren wir diejenigen Buttons, die später in benutzerdefinierte Objekte umgewandelt werden sollen, durch Bit 10 in den Objekt-Flags. Innerhalb des Programms wird der Objekttyp der so markierten Buttons vor dem Betreten des Dialogs durch den *G_USERDEF*-Typ ersetzt. Damit der Zeiger *ob_spec* der Buttons nicht verloren geht, wird er in der *USERBLK*-Struktur, die für jedes Objekt getrennt erzeugt werden muß, als Parameter *ub_parm* untergebracht. Anschließend kann der alte Inhalt von *ob_spec* durch einen Zeiger auf die entsprechende *USERBLK*-Struktur ersetzt werden. Sobald ein Button gezeichnet werden soll, wird nun vom GEM unsere eigene Zeichenroutine angesprungen, der als Parameter ein Zeiger auf eine *PARMBLK*-Struktur übergeben wird, die alle Angaben enthält, die wir für unser weiteres Vorgehen benötigen. Nach Beendigung des Dialogs wird der Ausgangszustand hergestellt, indem alle Objekte des Typs *G_USERDEF* wieder in *G_BUTTON*-Objekte umgewandelt werden.



Abb. 2: Beispieldialog aus der Sicht des RSC

```
typedef struct
{
     char *fd_addr;           /* Zeiger auf Speicherblock */
     int  fd_w;               /* Breite des Blocks in Punkten */
     int  fd_h;               /* Höhe des Blocks in Punkten */
     int  fd_wdwidth;         /* Breite des Blocks in Worten */
     int  fd_stand;           /* Formatflag (0: geräteabhängig
                                             1: Standardformat) */
     int  fd_nplanes;         /* Zahl der Bildebenen */
     int  fd_r1,fd_r2,fd_r3;  /* reserviert */
} MFDB;
```

Listing 4

Welches Zeichen soll markiert werden?

Bleibt nur noch die obige Frage: Wo wird die Information untergebracht, welches Zeichen unterstrichen werden soll? Auch hierzu empfiehlt es sich, bereits in den Resource-Daten eine entsprechende Angabe unterzubringen. Die hier verwendete Möglichkeit besteht darin, Bit 7 im ASCII-Code des Zeichens zu setzen, das später auf dem Bildschirm ein besonderes Textattribut erhalten soll. Zwar befindet sich nun innerhalb des RCS ein hieroglyphisch anmutendes Zeichen auf dem Bildschirm (Abbildung 2), aber wenn wir dieses Zeichen im Programmverlauf ausgeben, werden wir Bit 7 natürlich vorher zurücksetzen, so daß ein ganz normaler Text auf dem Bildschirm erscheint.

Maximale Flexibilität

Vielleicht mag es dem einen oder anderen unsinnig erscheinen, mit diesem recht ungewöhnlichen Verfahren zu arbeiten, aber so ist es möglich, wirklich alle Informationen, die für den erweiterten Dialog benötigt werden, in den Resource-Daten unterzubringen. Hierdurch erhält man eine große Flexibilität, was eventuelle Änderungen betrifft. Es genügt in diesem Fall, ausschließlich die Resourcen zu ändern, ohne das restliche Programm neu assemblieren oder compilieren zu müssen.

Auf die Taste kommt es an

Nachdem die eigentliche Darstellung der Buttons geklärt ist, geht es nun noch darum, an geeigneter Stelle eine Information darüber abzulegen, welche Taste in Verbindung mit der Alternate-Taste betätigt werden muß, um einen Knopf des Dialogs anzusprechen. Hierzu soll das gleiche Verfahren eingesetzt werden, das bereits in [1] angesprochen wurde. Der Scancode der Taste, die für einen bestimmten Button gedrückt werden muß, wird im high-Byte des Objekttyps *ob_type* untergebracht. Somit werden alle Informationen, die für einen erweiterten Dialog benötigt werden, in den Resource-Daten untergebracht und benötigen keinen zusätzlichen Speicherplatz in Form irgendwelcher Tabellen.

Schreiten wir also zur Tat

Nun sollten wir einen Blick ins Programm-Listing werfen.

Nach der obligatorischen GEM-Initialisierung finden wir die bereits in [1] vorgestellte erweiterte *form_do*-Routine, auf die hier nicht erneut eingegangen werden soll. Wichtig sind im momentanen Zusammenhang lediglich die Aufrufe der Unterprogramme *inituser* und *exituser*. inituser überprüft vor dem Zeichnen der Dialogbox für jedes Objekt des Dialogs, ob es sich um einen Button handelt, der in ein *G_USERDEF*-Objekt umgewandelt werden soll. Dies geschieht durch Überprü-

fung von Bit 10 des Objektstatus, das ja solche Objekte kennzeichnet. Bei Bedarf wird nun eine *USERBLK*-Struktur für diese Objekte aufgebaut. Das zweite Langwort dieser Struktur enthält den Pointer auf den Text des Buttons, also den *ob_spec*-Zeiger des Knopfes. (Damit der Strich unter dem markierten Zeichen besser sichtbar ist, wird der Rahmen des Knopfes übrigens um ein Pixel nach unten hin vergrößert.) exituser macht die von inituser vorgenommenen Änderungen wieder rückgängig, ersetzt also alle Objekte des Typs *G_USERDEF* durch *G_BUTTON*-Objekte.

Die Unterroutine usercode übernimmt schließlich das Zeichnen der selbstdefinierten Objekte. Dieses Unterprogramm wird vom AES aufgerufen, wenn ein Objekt neu gezeichnet werden muß. Dies ist auch dann der Fall, wenn sich der Objektstatus ändert, also das Objekt z.B. selektiert wurde. Innerhalb dieser Routine dürfen übrigens nur Aufrufe des VDI, jedoch keine AES-Aufrufe erfolgen. Als Parameter erhält *usercode* auf dem Stack einen Zeiger auf die *PARMBLK*-Struktur des aktuellen Objekts.

Durch Vergleich von *pb_prevstate* und *pb_currstate* stellt *usercode* fest, ob der Button zum ersten Mal gezeichnet wird, oder ob es sich lediglich um eine Statusänderung handelt. Sind alter und neuer Objektstatus identisch, wird das Objekt zum ersten Mal gezeichnet. In diesem Fall werden vom Programm zunächst alle benötigten Schreibattribute gesetzt (*vswr_mode*, *v_pline*). Anschließend werden die Koordinaten für den auszugebenden String berechnet, der ja, wie innerhalb von Buttons üblich, zentriert dargestellt werden soll. Die eigentliche Textausgabe geschieht mit Hilfe von *v_gtext*. Alle Zeichen, bei denen Bit 7 nicht gesetzt ist, werden in normaler Schrift ausgegeben. Bei dem von uns markierten Zeichen wird vor der Ausgabe Bit 7 gelöscht und mittels *vst_effects* das Textattribut "unterstrichen" gewählt.

Vor dem Verlassen von usercode wird der Objektstatus, mit dem der Button versehen ist, im Datenregister D0 an das AES übergeben. Dies ist deshalb notwendig, weil das AES anschließend selbständig alle notwendigen Operationen vornimmt, um das von uns ohne jegliche Statusinformationen dargestellte Objekt mit dem richtigen Status zu versehen. Der Programmierer braucht sich also nicht selber um die Verwaltung der Status-Bits zu kümmern!

Sind alter und neuer Objektstatus beim Betreten von usercode nicht gleich, befindet sich der Button bereits auf dem Bildschirm und wurde selektiert bzw. deselektiert. Es geht nun also darum, den Button zu invertieren. Hierzu wird die Rasterfunktion *vro_cpyfm* verwendet. Da diese Funktion nur recht selten eingesetzt wird, möchte ich ein wenig näher auf ihre Möglichkeiten eingehen.

Funktionsweise von vro_cpyfm

vro_cpyfm ermöglicht es, zwei rechteckige Bildschirmbereiche logisch miteinander zu verknüpfen. Dabei gibt es 16 Verknüpfungsmodi [2]:

Modus	Wirkung
0	E = 0
1	E = Q AND Z
2	E = Q AND (NOT Z)
3	E = Q
4	E = (NOT Q) AND Z
5	E = Z
6	E = Q XOR Z
7	E = Q OR Z
8	E = NOT(Q OR Z)
9	E = NOT(Q XOR Z)
10	E = NOT Z
11	E = Q OR (NOT Z)
12	E = NOT Q
13	E = (NOT Q) OR Z
14	E = NOT(Q AND Z)
15	E = 1

(Q = Pixelwert des Quellpixels
Z = Pixelwert des Zielpixels
E = Pixelwert des Zielpixels nach der Verknüpfung)

```
 1:  ********************************************
 2:  * Beispielprogramm für die Realisierung  *
 3:  * eines komfortablen Dialogs             *
 4:  * (C) MAXON Computer GmbH                *
 5:  * by Uwe Seimet                          *
 6:  ********************************************
 7:
 8:  *benötigte Objekttypen
 9:  G_USERDEF= 24
10:  G_BUTTON = 26
11:
12:  *wichtige Variablen OBJECT-Struktur
13:  ob_flags = 8
14:  ob_spec  = 12
15:
16:  *wichtige Variablen der PARMBLK-Struktur
17:  pb_prevstate = 6
18:  pb_currstate = 8
19:  pb_x      = 10
20:  pb_y      = 12
21:  pb_w      = 14
22:  pb_h      = 16
23:  pb_parm   = 26
24:
25:  *für einige wenige Systemaufrufe
26:  GEMDOS    = 1
27:  MSHRINK   = 74
28:
29:  BIOS      = 13
30:  BCONOUT   = 3
31:
32:
33:           text
34:
35:           move.l sp,a0
36:           lea stack+400,sp
               ;Stackpointer initialisieren
37:           move.l 4(a0),a0
38:           move.l 12(a0),a1
39:           add.l 20(a0),a1
40:           add.l 28(a0),a1
41:           lea $100(a1),a1          ;für Basepage
42:           pea (a1)
43:           pea (a0)
44:           clr -(sp)
45:           move #MSHRINK,-(sp)
46:           trap #GEMDOS
                              ;Restspeicher freigeben
47:           lea 12(sp),sp
48:           tst.l d0
49:           bne error                ;Fehler-
50:           lea intin,a5             ;Pointer auf
                                        INTIN-Array
51:           lea intout,a6            ;Pointer auf
                                        INTOUT-Array
52:           moveq #10,d0             ;appl_init
53:           move.l #$00010000,d1
54:           bsr aes
55:           bmi error                ;Fehler-
56:           moveq #77,d0             ;graf_handle
57:           move.l #$00050000,d1
58:           bsr aes
59:           move 2(a6),hwchar        ;Zeichenbreite
                                        in Pixeln
60:           move 4(a6),hhchar        ;Zeichenhöhe
                                        in Pixeln
61:           move.l a5,a0
62:           moveq #9,d0
63:  opn:     move #1,(a0)+
64:           dbra d0,opn
65:           move #2,(a0)
66:           moveq #100,d0            ;v_opnvwk
67:           lea contrl+2,a3
68:           clr.l (a3)+
69:           move #11,(a3)
70:           move (a6),6(a3)          ;graf_handle
71:           bsr vdiinit
72:           move 6(a3),vdi_h         ;vdi_handle
                                        merken
73:           move.l (a6),d0           ;Desktopgröße
```

Man sieht, daß nicht für alle Verknüpfungen zwei Bildschirmbereiche benötigt werden. Manche Operationen arbeiten auch mit einem einzigen Ausschnitt, der dann mit einem konstanten Wert verknüpft wird. (Die Rasterfunktionen werden bei Rechnern mit eingebautem Blitter übrigens vom Blitter zur Verfügung gestellt. Alle Operationen müssen sich nicht unbedingt auf den Bildschirm beziehen, sondern können auch auf andere Speicherbereiche angewandt werden.)

Zum Invertieren eines Bildschirmbereichs eignet sich Verknüpfungsmodus 12, bei dem lediglich jedes Pixel des Quellrasters invertiert wird, um das Zielraster zu erhalten. Neben der Art der Verknüpfung erhält *vro_cpyfm* als Parameter noch zwei sogenannte MFDBs (Memory Form Definition Blocks): (s.Listing 4)
Der MFDB versorgt das VDI somit mit einer näheren Beschreibung der zu verknüpfenden Speicherbereiche. Wird die logische Verknüpfung auf den Bildschirm angewandt, so kann man für *fd_addr* einen Null-Pointer einsetzen. In diesem Fall werden automatisch die korrekten Parameter für den Bildschirmspeicher (im gerätespezifischen Format) eingesetzt. Auch innerhalb der Routine usercode wird so verfahren. Da nur ein einziger Block von der Verknüpfung mit Modus 12 betroffen ist, kann derselbe MFDB für das Quell- und das Zielraster verwendet werden. Nach dem Aufruf von *vro_cpyfm* gibt usercode die Kontrolle wieder an das AES ab.

Soweit einige Erläuterungen zum kommentierten Assembler-Quelltext des Programms, dem weitere Einzelheiten entnommen werden können. Nach dem Assemblieren und Linken sollte die Programmlänge übrigens 1712 Bytes betragen. Je nach verwendetem Assembler können hier kleinere Abweichungen auftreten.

Grenzen des vorgestellen Verfahrens

An sich kann man mit Hilfe von benutzerdefinierten Objekten jede nur denkbare Form von Objekten erzeugen. Sind diese Objekte jedoch nicht

```
 74:           add.l #$00010001,d0
 75:           move.l d0,deskwh
                          ;Bildschirmbreite und -höhe
 76:           lea _objcnr,a0
 77:           move a0,(a5)
 78:           bra.s skip
 79:  fixloop: moveq #114,d0               ;rsrc_obfix
 80:           move.l #$01010100,d1
 81:           lea objc000,a0
 82:           bsr aesobj                  ;Objektdaten
                                            umrechnen
 83:  skip:    subq #1,(a5)
 84:           bpl fixloop
 85:           lea objc000,a2
 86:           clr d5                      ;kein
                                            Eingabefeld
 87:           bsr.s form_do               ;Dialog
                                            ausführen
 88:           moveq #101,d0               ;v_clsvwk
 89:           bsr vdi
 90:           moveq #19,d0                ;appl_exit
 91:           move.l #$00010000,d1
 92:           bsr aes
 93:  error:   clr -(sp)
 94:           trap #GEMDOS
 95:
 96:  form_do:
 97:           clr (a5)                    ;Pfeil als
                                            Mauscursor
 98:           moveq #78,d0                ;graf_mouse
 99:           move.l #$01010100,d1
100:           bsr aes
101:           move.l a2,a0
102:           moveq #54,d0                ;form_center
103:           move.l #$00050100,d1
104:           bsr aesobj
105:           movem.l 2(a6),d0/d1         ;form_xy und
                                            form_wh
106:           movem.l d0/d1,xywh
107:           clr d2                      ;FMD_START
108:           movem.l d0/d1,2(a5)
109:           movem.l d0/d1,10(a5)
110:           bsr form_dial
111:           moveq #3,d0                 ;bis zu
                                            Ebene 3
112:           move.l d0,(a5)
113:           bsr inituser     ;G_USERDEF-
                                 Objekte initialisieren
114:           movem.l xywh,d0/d1
115:           movem.l d0/d1,4(a5)
116:           moveq #42,d0                ;objc_draw
117:           move.l #$06010100,d1
118:           move.l a2,a0
119:           bsr aesobj
120:           bra _form_do
121:
122:  form_cnt:
123:           bsr exituser       ;G_BUTTON-
                                   Objekte zurückholen
```

```
124:           moveq #3,d2        ;FMD_FINISH
125:           movem.l xywh,d0/d1
126:           movem.l d0/d1,2(a5)
127:           bsr.s fo_dial
128:           move d7,d0
129:           mulu #24,d0
130:           move 10(a2,d0),d0        ;alter
                                         Objektstatus
131:           bclr #0,d0
                                        ;deselektieren
132:           clr d1                   ;nicht neu
                                         zeichnen
133:           move.l a2,a0
134:  change:  move d7,d2
135:           mulu #24,d2
136:           tst.b 9(a0,d2)          ;versteckt?
137:           bpl *+4                 ;nein-
138:           clr d1                  ;nicht neu
                                        zeichnen
139:           move 10(a0,d2),d2       ;alter
                                        Objektstatus
140:           and #$fff6,d2
141:           or d2,d0
142:           move.l a5,a1
143:           move d7,(a1)+               ;Objektnummer
144:           clr (a1)+
145:           move.l deskxy,(a1)+
146:           move.l deskwh,(a1)+         ;Breite und
                                            Höhe
147:           move d0,(a1)+               ;neuer
                                            Objektstatus
148:           move d1,(a1)                ;Flag für
                                            neu zeichnen
149:           moveq #47,d0                ;objc_change
150:           move.l #$08010100,d1
151:           bra aesobj
152:
153:  form_end:
154:           moveq #3,d2                 ;FMD_FINISH
155:           movem.l xywh,d0/d1
156:           movem.l d0/d1,2(a5)
157:  fo_dial: movem.l xywh,d0/d1
158:           movem.l d0/d1,10(a5)
159:  form_dial:
160:           moveq #51,d0                 ;form_dial
161:           move.l #$09010100,d1
162:           move d2,(a5)
163:           bra aes
164:
165:  _form_do:movem d5-d6,-(sp)
166:  _form:   moveq #1,d0             ;Cursor
                                        einschalten
167:           bsr objcedit
168:           move 2(a6),d6           ;Position
                                        des Cursors
169:  dloop:   move #2,2(a5)
170:           move #1,4(a5)           ;linker
                                        Mausbutton
```

mehr rechteckig, sondern z.B. rund, wird der Programmieraufwand größer. In solchen Fällen ist es nämlich nicht mehr möglich, den Objektstatus durch das AES verwalten zu lassen. Von Icons einmal abgesehen, werden vom AES ausschließlich rechteckige Objekte unterstützt. Will man andere Objektformen einsetzen, muß man also nicht nur das eigentliche Zeichnen des Objekts beim ersten Aufruf des Dialogs selber vornehmen, sondern auch alle Routinen zur Verfügung stellen, um auf Änderungen des Objektstatus' zu reagieren, was natürlich einen erhöhten Aufwand mit sich bringt.

Literatur:

[1] "form_do-Routine - Eingabe einmal anders", ST-Computer 1/90

[2] Jankowski, Reschke, Rabich, "ATARI ST Profibuch", Sybex-Verlag



```
171:          move #1,6(a5)           ;gedrückter
                                       Mausbutton
172:          moveq #25,d0            ;evnt_multi
173:          move.l #$10070100,d1
174:          move #3,(a5)            ;MU_KEYBD |
                                       MU_BUTTON
175:          lea ev_buff,a0
176:          clr (a0)                ;letzte
                                       Message löschen
177:          bsr aesobj
178:          btst #1,1(a6)           ;MU_BUTTON?
179:          bne.s button            ;ja-
180:          move 10(a6),d0          ;ASCII- und
                                       Scancode
181:          tst.b d0                ;Alternate-
                                       Taste gedrückt?
182:          bne.s noalt             ;nein-
183:          lsr #8,d0
184:          move.l a2,a0
185: uloop:   cmp.b 6(a0),d0          ;Button für
                                       Taste gefunden?
186:          beq.s default           ;ja-
187:          btst #5,9(a0)           ;letztes
                                       Objekt?
188:          bne.s noalt             ;ja-
189:          lea 24(a0),a0
190:          bra uloop
191: default: tst.b 9(a0)             ;verstecktes
                                       Objekt?
192:          bmi.s noalt             ;ja-
193:          sub.l a2,a0
194:          move.l a0,d7
195:          divu #24,d7             ;ergibt
                                       Objektnummer
196:          move #1,12(a6)          ;für einen
                                       Mausklick
197:          bra.s obj
198: noalt:   move 10(a6),2(a5)
                                      ;eingegebenes Zeichen
199:          move d5,(a5)            ;aktuelles
                                       Edit-Objekt
200:          clr 4(a5)
201:          moveq #55,d0                 ;form_keybd
202:          move.l #$03030100,d1
203:          move.l a2,a0
204:          bsr aesobj
205:          clr d7
206:          move.b 3(a6),d7              ;neues Edit-
                                            Objekt
207:          tst (a6)                     ;Exit?
208:          beq exit                     ;ja-
209:          tst 4(a6)                    ;Wechsel des
                                            Eingabefeldes?
210:          beq.s newin                  ;ja-
211:          move 10(a6),2(a5)
                                      ;eingegebenes Zeichen
212:          move d6,4(a5)
                                      ;Cursorposition
```

```
213:          moveq #2,d0         ;Zeichen ausgeben
214:          bsr objcedit
215:          move 2(a6),d6       ;neue Cursorposition
216:          bra dloop
217: button:  clr (a5)
218:          move #3,2(a5)       ;bis Ebene 3
219:          move.l 2(a6),d0     ;aktuelle
                                   Mausposition
220:          move.l d0,4(a5)
221:          moveq #43,d0        ;objc_find
222:          move.l #$04010100,d1
223:          move.l a2,a0
224:          bsr aesobj
225:          move (a6),d7        ;Objektnummer
226:          bpl.s obj           ;Mauszeiger
                                 ist innerhalb der Box-
227:          pea (a2)
228:          bsr bell            ;Warnton ausgeben
229:          move.l (sp)+,a2
230:          bra dloop
231: obj:     moveq #56,d0        ;form_button
232:          move.l #$02020100,d1
233:          move d7,(a5)
                                  ;angeklicktes Objekt
234:          move 12(a6),2(a5)  ;Anzahl der
                                   Mausklicks
235:          move.l a2,a0
236:          bsr aesobj
237:          tst (a6)                      ;Exit-Objekt?
238:          beq.s exit                    ;ja-
239:          move d7,d0
240:          mulu #24,d0
241:          btst #3,9(a2,d0)              ;Objekt
                                             edierbar?
242:          beq dloop                     ;nein-
243:          cmp d7,d5
244:          beq dloop
245: newin:   bsr.s cursor
246:          move d7,d5                  ;neues
                                           Eingabefeld
247:          bra _form
248: exit:    bsr.s cursor
249:          movem (sp)+,d5-d6
250:          bra form_cnt
251: cursor:  move d6,4(a5)            ;Cursorposition
252:          moveq #3,d0
253: objcedit:move d5,(a5)             ;Eingabefeld
                                        vorhanden?
254:          bne *+4                  ;ja-
255:          rts
256:          move d0,6(a5)            ;Cursor
                                        einschalten
257:          moveq #46,d0             ;objc_edit
258:          move.l #$04020100,d1
259:          move.l a2,a0
260:          bra aesobj
261:
262: *markierte Objekte in G_USERDEF Objekte umwandeln
263: *USERBLK-Struktur erzeugen
264: inituser:
265:          lea buffer,a1                 ;Puffer für
                                      USERBLK-Strukturen
266:          move.l a2,a0
267:          clr d0
268: iloop:   btst #2,ob_flags(a0,d0)      ;markiertes
                                            Objekt?
269:          beq.s noext1                 ;nein-
270:          move.l ob_spec(a0,d0),d1     ;Pointer auf
                                            Text
271:          move.l a1,12(a0,d0)          ;Pointer auf
                                        USERBLK-Struktur
272:          move.l #usercode,(a1)+       ;Pointer auf
                                   eigene Zeichenroutine
273:          move.l d1,(a1)+              ;Pointer auf
                                            Text
274:          move.b #G_USERDEF,7(a0,d0)
275:          move hhchar,22(a0,d0)
276:          addq #1,22(a0,d0)            ;Button um
                                    ein Pixel erweitern
277: noext1:  add #24,d0
278:          btst #5,-15(a0,d0)           ;LASTOB?
279:          beq iloop                    ;nein-
280:          rts
281:
282: *G_USERDEF-Objekte in G_BUTTON-Objekte
      zurückverwandeln
283: exituser:
```



Ladenlokal: Gustav-Mahler-Straße am Einkaufszentrum
viele Parkplätze direkt am Laden

Weide Elektronik GmbH
Regerstr. 34
D-4010 Hilden
Telefon 0 21 03 / 4 12 26
Telefax 0 21 03 / 3 18 20

ATARI System-fachhändler
Leasing für Komplettsysteme möglich
Atari + PC XT/AT

Schweiz
DTZ Data Trade AG
Landstr. 1
CH-5415 Rieden - Baden
Tel. (056) 82 18 80
Fax (056) 82 18 84

Österreich
Reinhart Temmel GmbH
Markt 109
A-5440 Golling
Tel. (0 62 44) 70 81-0
Telefax 7 18 83

STECHUHR ST

Manipulationsgeschützte Software zur Erfassung von Arbeitszeiten mittels Kennungsdiskette für jeden Mitarbeiter

Nutzen

- verläßliche Stundenlisten
- individuelle Lohngestaltung und Lohnabrechnung
- Kostensenkung durch Zeitersparnis und exaktes Datenmaterial

Leistungsmerkmale

- Sekundengenau erfaßte Ankunfts- und Abgangszeiten
- einfache Erfassung durch Einschieben der für jeden Mitarbeiter spezifischen Kennungs-Diskette
- Entnahme der Protokolldiskette akustisch gesichert
- Datum und Uhrzeit werden automatisch über die interne Uhr des Rechners erfaßt
- exakt berechnete Arbeitsstunden pro Mitarbeiter
- beliebig wählbare Auswertungszeiträume
- Protokoll der täglichen Arbeitszeiten auf Diskette und Drucker (optional)
- automatische Fehlerkorrektur bei vergessener Erfassung
- MS-DOS kompatibles Diskettenformat
- der Rechner ist weiterhin für andere Zwecke einsetzbar, falls als ACC benutzt
- Zeiten können für Lohnabrechnung gewichtet werden
- individuelle Lohnabrechnung für jeden Mitarbeiter durch
- eigene Abrechnungsformel mit zeit- und lohnabhängigen Faktoren
- komplexe Berechnungen in Abrechnungsformularen

ADRESS DIREKT

Datensicherheit und Komfort

- geschützt durch Passwort
- Schutz der Installation vor Manipulationen wie Löschen etc.
- 'private' Datensätze speicherbar, d.h. diese sind nur für den Anwender zugänglich, der sie gespeichert hat - der Superuser kann alle sehen
  beliebig viele Installationen speicherbar
- Zugriffsrechte für 'Speichern', 'Ändern', 'Löschen',
- 'private Datensätze', 'Drucken', 'Listenbearbeitung', 'Installationserlaubnis' einzeln und für jeden Benutzer getrennt einstellbar
- jeder Installation kann ein eigener Pfad für den Dateizugriff vergeben werden
- Adressdaten nicht in einer Datei, sondern in mehreren verteilt, dadurch bei Diebstahl nicht reproduzierbar
- viele Einzeldaten kodiert abgespeichert
- jeder Installation, das heißt jedem Benutzer, kann eine eigene Textverarbeitung zugeordnet werden
- Start beliebiger Fremdprogramme möglich
- bis zu 7 Fenster mit beliebigen Inhalten
- großbildschirmfähig
- Zugriff auf die installierten ACC's
- bequemes Fensterhandling auch über Tastatur
- weit über den GEM-Standard hinausgehende Möglichkeiten der Dialogbearbeitung
- Popup-Menüs mit den wichtigsten Funktionen
- sofortiges Erkennen von Falscheingaben
- sofortige Feldformatierungen mit intelligenten Ergänzungsroutinen für Datum/Zeit
  editierbare Dateilisten
  in Listen geänderte bzw. gelöschte DS hervorgehoben
- Dateilisten abspeicherbar und wieder ladbar (stets mit aktuellen Werten!)
- schnelle Textsuche auch in Dateilisten unter Verwendung von Wildcards
- mächtige Listenfunktionen mit Rechnen im Text,
- Platzhalter für Datum, Uhrzeit, vielseitige Formatiermöglichkeiten
- verschachtelte Listenausgabe möglich
- schnelle Listenausgabe auf Laserdrucker, bei extrabreiten Listen auch gedreht
- flexibler Import (z.B. von BS-Handel-Daten, 1st-Address, Adimens, DBase)
- Netzwerkfähige Version für das PAM's Net lieferbar

XCOPY.DFÜ....ein Muß für alle Anwender mit mehr als einem ST – DFÜ-fähiges Kopierprogramm für RS232-, MIDI- und parallele Schnittstelle mit integr. Terminal und Kontrollfeld. Einsatzgebiet: Schneller Datentransfer von Rechner zu Rechner ohne Diskette; Anpassen der Inhalte versch. Festplatten an versch. Arbeitsplätzen bei automat. Sicherung bereits vorh. Daten; umfangreiche Selektionsmöglichkeiten der zu kopier. Dateien: Wildcards, Archivbit, autom. Backup etc; Versenden von Mitteil. von ST zu ST (Progr. arbeitet im Hintergrund) Einsetzbar als ACC, PRG (auch mit Parametern von Shells oder and. Programmen. Bitte INFO anfordern!

COMPUTER-PROJEKTIONSPANEL

Für alle, die etwas präsentieren müssen!

- Gestochen scharf, 8 Graustufen, hochmodernes DST
- LCD mit IR Fernbedienung!
- Keine Spezialsoftware nötig.
- Geeignet für jeden Overhead Projektor.
- Invertier-Funktion, Löschfunktion, auch an IBM PS/2, IBM PC XT/AT mit CGA, EGA, VGA-Karte an schließbar!
- Auflös. 640 x 480 Bildpunkte, nur 3 kg Gewicht!

Passender Metallkoffer 348,- **Preis 3798,-**

SOFTWARE

CAD

Cadja	998,-
Technobox Drafter	798,-
Technobox CAD/2	1998,-

Vorführung von kompletten CAD-Systemen unter MS-DOS oder auf Atari ST in unserem Beratungszentrum. Rufen Sie an!

DTP

Calamus	798,-
Calamus Font Editor	198,-
Outline Art	378,-
PKS-Write	198,-
Didot Fonteditor f. Calamus	198,-

Wir bieten Ihnen als DTP-Center individuelle Beratung für den Systemkauf im Bereich Desktop-Publishing. Rufen Sie an!

Textverarbeitung

1st Word Plus V3.15 inkl. 1st Address u. 1st Extra	239,-
1st Proportional Plus	119,-
Signum II	438,-
Scarabus Signum Fonteditor	98,-
Daily Mail	179,-
Writer ST	148,-
Word Perfect ST	748,-
That's Write Junior	148,-
That's Write	328,-
That's Write PostScript	498,-
Tempus 2.0	129,-
Tempus Word	a.A.

Graphik

Creator	229,-
Imagic	458,-
Megapaint II	449,-
Megapaint II prof.	798,-
Omikron DRAW! 3.0	119,-
STAD	169,-
Retouche	399,-
Arabesque	268,-
Lavadraw	149,-

Datenbanken

1st Address	148,-
Adimen's	389,-
dbMAN V Version 5.2 incl. Comp., ISI.+Tempus	978,-

DFÜ

BTX-Manager incl. Interface	398,-
ohne Interface	278,-
Interlink ST	79,-

Tabellenkalkulation

BasiCalc	98,-
BasiChart	198,-
JAMES Das Börsenprogramm	298,-
GFA-Statistik	948,-

Kaufm. Software

Software BS-FiBu	548,-
Software BS-Handel	498,-
Software BSS Plus System	a.A.
Software BSS Plus System (netzwerkfähig GTI-Netz)	a.A.

Programmiersprachen

GFA-ASSEMBLER ST	139,-
Omikron Assembler	94,-
Omikron Macro-Assembler	
GFA-Basic 2.0 EWS ST	44,90
GFA-Basic 3.0 EWS ST	189,-
GFA-Basic 3.5 EWS ST 40 neue Befehle (Vektor/Matrix)	268,-
GFA-Basic 68881 EWS ST	268,-
Omikron Basic	14,90
Omikron Basic 68881 Comp.	219,-
Omikron Basic Compiler	169,-
Omikron Basic Libraries	
diverse Libraries	ab 89,-
Omikron Basic Lib. SQL	288,-
Laser C	388,-
Prospero C	248,-
Turbo C	229,-
Turbo C + Debugger	448,-
Prospero Fortran	378,-
Prospero Fortran 68881 Lib.	198,-
SPC Modula-2	348,-
Prospero Pascal	248,-
GFA-BASIC KONVERTER nach C	478,-
GFA-CUP GEM UTILITY-PACKAGE	139,-

MIDI

Steinberg Cubase	790,-
Steinberg Twelve	99,-
EMC-Soundeditoren für verschied. Synthesizer/ Expander	
Roland D10/20/110	120,-
Ensoniq VFX	150,-
andere	a.A.

Utilities

Harlekin "ein ACC für alle Fälle	129,-
Turbo ST" Der Softwareblitter macht dem ST Beine	79,-
Querdruck " universell anwendbar für Tabellenkalkulation	79,-
BOOT IT " DAS Startprogramm für Festplatten	69,-

HARDWARE

Atari Mega ST1 mit 4 MB + SM124 + Maus	2998,-

Weide SCSI-Festplatten

Weide HD 50, 28 ns	1498,-
Weide HD 65, 28 ns	1698,-
Weide HD 85, 28 ns	1898,-
Weide HD 111, 19 ns Cache	2298,-
Weide HD 132, 19 ns Cache	2548,-
Weide HD 172, 19 ns Cache	3198,-
größere Platten	a.A.

Alle Festplatten mit umfangreichen HD-Utilities, 100% komp. zu Atari-Festplatten, stabiles Metallgehäuse, ohne Lüfter!

Streamer

ICD-Streamer 155 MB	2498,-

Komfortable, schnelle u. zuverlässige Sicherung Ihrer Daten auf eine Datenkassette. Datensicherung mit 6.5 MB/Minute, durchgeführter und gepufferter DMA- und SCSI-Bus

Sub-Systeme

("HD + Streamer in einem Gehäuse)

Weide HD 50 + Streamer	3798,-
Weide HD 111 + Streamer	4598,-
Weide HD 132 + Streamer	4998,-
Weide HD 172 + Streamer	5998,-

Floppy

3.5", anschlußfertig	289,-
5.25", 40/80 Tracks	448,-

Plotter

DIN-A3, Flachb., 8 Stifte	2451,-
DIN-A4 bis DIN-A0	a.A.

u.a. Graptec A0-Plotter vorführbereit in unserem Beratungszentrum. Rufen Sie an!

Scanner

Printtechnik 200 dpi	948,-
Handyscanner 400 dpi, 105 mm	548,-
Bildverarbeitung auf Atari ST	
Epson GT-4000 Farbscanner 8 Bit, professionell	4998,-

Emulatoren

PC-Speed	498,-
Supercharger	598,-
Spectre	798,-
Spectre GCR	1298,-

Modem

Discovery 2400 C	448,-
Discovery 2400 P Pocketmodem	449,-

Der Anschluß und Betrieb der Modems

Weide Speicherkarten, die bewährten!

auf 1.0 MB für Atari 260, 520	348,-
auf 2.5 MB für Atari 260, 520, 1040	898,-
auf 2.5 MB für Atari Mega ST1	948,-
auf 4.0 MB für Atari Mega ST1	1548,-
auf 4.0 MB für Atari 260, 520, 1040	1498,-

Jede Erweiterung einzeln im Rechner getestet. Bebil-derte Einbauanleitung, einfacher Einbau ohne Löten. Durch vergoldete Mikrosteckkontakte optimale Schonung des MMU-Sockels. Kein Bildschirmflimmern, keine zus.Software, keine zus. Stromversorgung

am Netz der DBP ist strafbar!

Echtzeituhr Plus 129,-

Ab TOS '87 keine Boot-Software nötig. Jede Uhr im Rechner getestet und gestellt. Interner Einbau ohne Löten. Dadurch freier ROM-Port. Schaltjahrerkennung, Datum, Uhrzeit. Tausendfach bewährt!

Weide Coprozessor 68881

f. Mega ST incl. Software	498,-
Weide Coprozessor 68881 + WCL	898,-

Weide Coprozessor Language bringt Höchstleistung mit dem 68881 Coproz. INFO anfordern!

ICD Host Adapter

für Festplatten im Selbstbau
Anschluß von SCSI-Festplatten an Atari ST.
Voller SCSI-Befehlssatz verfügbar.
Umfangreiche Harddisk-Utilities.

Micro ST, die kompakteste Lösung für den Anschluß einer SCSI-Festplatte	249,-
Advantage ST, mit gepuffertem, durchgeschleiftem DMA-Bus	269,-
Advantage Plus ST mit zusätzl. Echtzeituhr	298,-

Digitizer

SAM-Echtzeit-Digitizer	998,-
Turbo-Dizer	348,-

Prommer

GAL-Prommer v. Maxxon	185,-
Junior Prommer	229,-
Rocke Eprommer	239,-

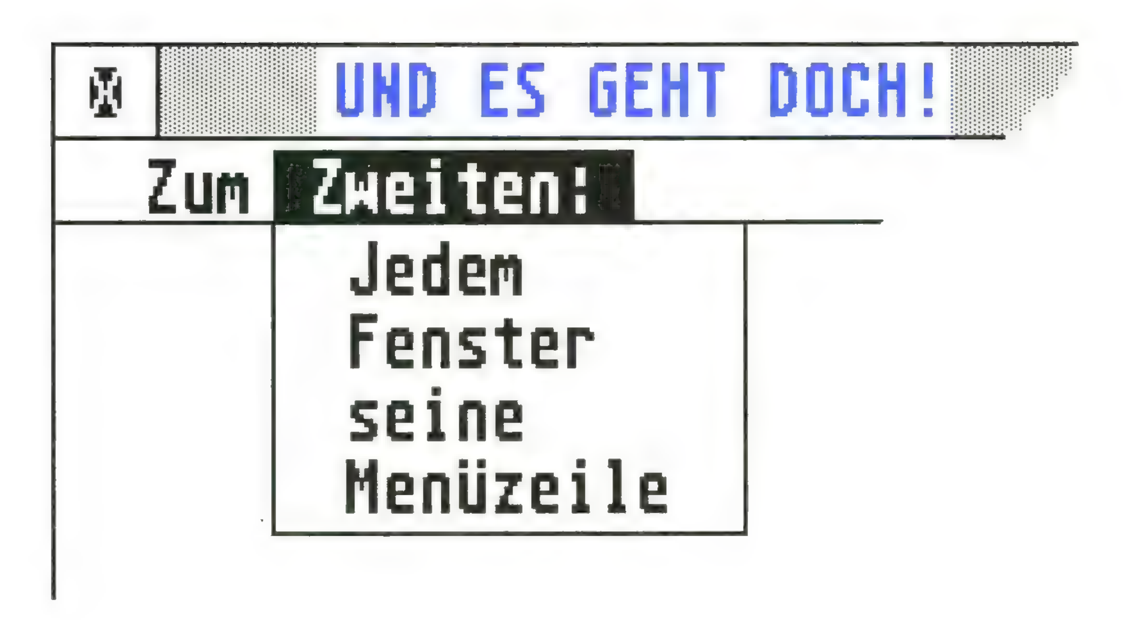
SIMM Speichererw. f. Atari STE

auf 2.0 MB	598,-
auf 4.0 MB	1198,-
Blitter Nachrüstplatine	89,-
Eprombank 64 KB	19,80
Epromdisk 512 KB	139,-
Atari Trackball f. Atari ST	98,-
kabellose Infrarot-Maus	198,-
Marconi Trackball, d. Professionelle	198,-

```
284:            lea buffer,a1        ;Puffer für
                                      alte ob_spec-Zeiger
285:            move.l a2,a0
286:            clr d0
287: eloop:     btst #2,ob_flags(a0,d0)
288:            beq.s noext2
289:            addq.l #4,a1
290:            move.l (a1)+,12(a0,d0)
291:            move.b #G_BUTTON,7(a0,d0)
292: noext2:    add #24,d0
293:            btst #5,-15(a0,d0)       ;LASTOB?
294:            beq eloop                ;nein-
295:            rts
296:
297:
298: usercode:
299:            movem.l a5/a6,-(sp)
300:            lea ptsin,a5
301:            move.l 12(sp),a6         ;Zeiger auf
                                          PARMBLK
302:            move 8(a6),d0
303:            cmp 6(a6),d0             ;erster
                                          Aufruf?
304:            beq.s first              ;ja-
305:            movem pb_x(a6),d0/d1
306:            movem d0/d1,(a5)
307:            movem d0/d1,8(a5)
308:            add pb_w(a6),d0
309:            subq #1,d0
310:            add pb_h(a6),d1
311:            subq #1,d1
312:            movem d0/d1,4(a5)
313:            movem d0/d1,12(a5)
314:            lea contrl,a0
315:            move #4,2(a0)
316:            move #1,6(a0)
317:            move.l #mfdb,14(a0)
318:            move.l #mfdb,18(a0)
319:            move #12,intin           ;E = not Q
320:            moveq #109,d0            ;vro_cpyfm
321:            bsr vdi                  ;Button
                                          invertieren
322:            move pb_currstate(a6),d0
323:            movem.l (sp)+,a5/a6
324:            rts
325: first:     moveq #32,d0             ;vswr_mode
326:            move.l #$00010001,contrl+6
327:            move #1,intin            ;REPLACE-
                                          Modus
328:            bsr vdi
329:            movem pb_x(a6),d0/d1
330:            subq #1,d0
331:            subq #1,d1
332:            movem d0/d1,(a5)
333:            movem d0/d1,16(a5)
334:            addq #1,d0
335:            add pb_w(a6),d0
336:            movem d0/d1,4(a5)
337:            addq #1,d1
338:            add pb_h(a6),d1
339:            movem d0/d1,8(a5)
340:            subq #1,d0
341:            sub pb_w(a6),d0
342:            movem d0/d1,12(a5)
343:            move #5,contrl+2
344:            moveq #6,d0              ;v_pline
345:            bsr vdi                  ;Rahmen
                                          zeichnen
346:            move.l pb_parm(a6),a4    ;Pointer auf
                                          String
347:            moveq #-1,d1
348: len:       addq #1,d1
349:            tst.b (a4,d1)            ;Stringlänge
                                          ermitteln
350:            bne len
351:            mulu hwchar,d1           ;ergibt
                               Stringbreite in Pixeln
352:            move pb_w(a6),d0
353:            sub d1,d0
354:            lsr #1,d0
355:            add pb_x(a6),d0
356:            move pb_y(a6),d1
357:            addq #1,d1
358:            movem d0/d1,(a5)         ;zentrierte
                                Koordinaten eintragen
359:            move hhchar,d0
360:            lsr #1,d0
361:            add d0,2(a5)
362:            lsr #1,d0
363:            add d0,2(a5)                   ;dto
364:            lea intin,a1
365:            clr d0
366:            moveq #-1,d1
367: lop:       addq #1,d1
368:            move.b (a4)+,d0
369:            move d0,(a1)+
370:            tst.b d0                 ;markiertes
                                          Zeichen erreicht?
371:            bpl lop                  ;nein-
372:            tst d1                   ;markiertes
                                 Zeichen erstes Zeichen?
373:            beq.s cont               ;ja-
374:            moveq #0,d0              ;Zeichen normal
375:            bsr.s gtext              ;bis zum
                               markierten Zeichen ausgeben
376: cont:      move.b -1(a4),d0
377:            and #$7f,d0              ;Bit 7 löschen,
                                          ergibt ASCII-Code
378:            move d0,intin
379:            moveq #1,d1              ;ein Zeichen
                                          ausgeben
380:            moveq #8,d0              ;Zeichen
                                          unterstrichen
381:            bsr.s gtext              ;markiertes
                                          Zeichen ausgeben
382:            lea intin,a1
383:            clr d0
384:            moveq #-1,d1
385: .loop1:    addq #1,d1
386:            move.b (a4)+,d0
387:            move d0,(a1)+
388:            bne.s .loop1             ;Stringende suchen
389:            tst d1                   ;folgen noch
                                          Zeichen?
390:            beq.s no                 ;nein-
391:            moveq #0,d0              ;Zeichen normal
392:            bsr.s gtext              ;restliche
                                          Zeichen ausgeben
393: no:        move pb_currstate(a6),d0
394:            movem.l (sp)+,a5/a6
395:            rts
396:
397: *Text ausgeben
398: gtext:
399:            move d1,-(sp)                  ;Stringlänge
400:            move intin,-(sp)
401:            move d0,intin                  ;Texteffekt
402:            move.l #$00010001,contrl+6
403:            moveq #106,d0                  ;vst_effects
404:            bsr.s vdi
405:            move (sp)+,intin
406:            move (sp),contrl+6
407:            move #1,contrl+2
408:            moveq #8,d0                    ;v_gtext
409:            bsr.s vdi
410:            move (sp)+,d0
411:            mulu #8,d0
412:            add d0,(a5)                    ;nächste
                                 Zeichenposition merken
413:            rts
414:
415: *AES aufrufen
416: aesobj:
417:            move.l a0,addrin
418: aes:       lea contrl,a0
419:            move d0,(a0)
420:            movep.l d1,3(a0)
421:            move.l #aespb,d1
422:            move #$c8,d0
423:            trap #2
424:            tst (a6)
425:            rts
426:
427: *VDI aufrufen
428: vdi:
429:            move vdi_h,contrl+12
430: vdiinit:   move d0,contrl
431:            move.l #vdipb,d1
432:            moveq #$73,d0
433:            trap #2
434:            rts
435:
436: *Warnton ausgeben
437: bell:
```

```
438:          move #$07,-(sp)          ;BEL
439:          move #2,-(sp)
440:          move #BCONOUT,-(sp)
441:          trap #BIOS      ;Warnton ausgeben
442:          addq.l #6,sp
443:          rts
444:
445:
446:  aespb:   dc.l contrl,global,intin,intout,addrin,
                    addrout
447:
448:  vdipb:   dc.l contrl,intin,ptsin,intout,ptsout
449:
450:  ****************************************
451:
452:  BOX      = 20
453:  TEXT     = 21
454:  BOXTEXT  = 22
455:  IMAGE    = 23
456:  USERDEF  = 24
457:  IBOX     = 25
458:  BUTTON   = 26
459:  BOXCHAR  = 27
460:  STRING   = 28
461:  FTEXT    = 29
462:  FBOXTEXT = 30
463:  ICON     = 31
464:  TITLE    = 32
465:
466:
467:  objc000:.dc.w $ffff
468:          .dc.w $0001,$0004
469:          .dc.w BOX
470:          .dc.w $0000,$0010
471:          .dc.l $00021100
472:          .dc.w $000a,$0003
473:          .dc.w $0022,$0005
474:
475:          .dc.w $0002
476:          .dc.w $ffff,$ffff
477:          .dc.w $781a
478:          .dc.w $0411,$0001
479:          .dc.l spec000            ;"Radiobutton Ö"
480:          .dc.w $0002,$0001
481:          .dc.w $000f,$0001
482:
483:          .dc.w $0003
484:          .dc.w $ffff,$ffff
485:          .dc.w $791a
486:          .dc.w $0411,$0000
487:          .dc.l spec001            ;"Radiobutton Ø"
488:          .dc.w $0012,$0001
489:          .dc.w $000e,$0001
490:
491:          .dc.w $0004
492:          .dc.w $ffff,$ffff
493:          .dc.w BUTTON
494:          .dc.w $0007,$0000
495:          .dc.l spec002            ;"OK"
496:          .dc.w $0008,$0003
497:          .dc.w $0009,$0001
498:
499:          .dc.w $0000
500:          .dc.w $ffff,$ffff
501:          .dc.w $611a
502:          .dc.w $0025,$0000
503:          .dc.l spec003            ;"ABBRUCH"
504:          .dc.w $0012,$0003
505:          .dc.w $0009,$0001
506:
```

```
507:  spec000:.dc.b "Radiobutton  ",0
508:
509:  spec001:.dc.b "Radiobutton  ",0
510:
511:  spec002:.dc.b "OK",0
512:
513:  spec003:.dc.b "ABBRUCH",0
514:
515:
516:
517:  ******************************
518:  * Konvertiert durch RSCCONV *
519:  *     Autor: Uwe Seimet     *
520:  *                           *
521:  ******************************
522:
523:  *Gesamtzahl an Objekten: 0005
524:  *Objektbäume: 0001
525:  *Tedinfos: 0000
526:  *Icons: 0000
527:  *Bitblöcke: 0000
528:
529:  *freie Strings: 0000
530:  *freie Images: 0000
531:
532:  _objcnr  equ 0005
533:
534:  ***********************************
535:
536:          bss
537:
538:  *die obligatorischen GEM-Arrays
539:
540:  contrl:  ds.w 11
541:
542:  global:  ds.w 15
543:
544:  intin:   ds.w 64
545:
546:  ptsin:   ds.w 64
547:
548:  intout:  ds.w 64
549:
550:  ptsout:  ds.w 64
551:
552:  addrin:  ds.w 64
553:
554:  addrout: ds.w 64
555:
556:  vdi_h:   ds.w 1   ;VDI-Handle
557:
558:  hwchar:  ds.w 1   ;Zeichenbreite in Pixeln
559:
560:  hhchar:  ds.w 1   ;Zeichenhöhe in Pixeln
561:
562:  deskxy:  ds.l 1
563:  deskwh:  ds.l 1   ;Breite und Höhe des Desktop
564:
565:  xywh:    ds.l 2   ;GröPe der aktuellen Dialogbox
566:
567:  ev_buff: ds.w 8   ;Message-Puffer
568:
569:  mfdb:    ds.w 18  ;Memory Form Descriptor Block
570:
571:  buffer:  ds.l 4   ;Puffer für USERBLK-Struktur
                       von zwei Buttons
572:
573:  stack:   ds.l 100 ;für eigenen Stack
```



Wie schon in dem Artikel "Submenüs unter GEM" (siehe [1]) soll auch hier wieder demonstriert werden, daß man mit "offiziellen" und "legalen" Mitteln unter GEM wesentlich mehr erreichen kann, als es sich die Entwickler von GEM vielleicht haben träumen lassen. Wie der Titel schon verrät, spielen auch dieses Mal die Menüs bzw. die Menüzeile wieder eine große Rolle; es soll hier ein Problem beseitigt werden, das auf anderen Rechnern schon lange gelöst wurde und eigentlich auch gar kein (oder genauer gesagt: kein so großes) Problem ist, wie wir gleich sehen werden.

Ein Beispiel
(direkt aus dem Leben gegriffen):

Welcher Leser kennt nicht folgende Situation: Hans-Otto sitzt vor seinem nagelneuen und großen (sehr großen) Großmonitor und arbeitet mit seinem ebenfalls neuen Programm Marke "Alleskönner". Dazu hat er diverse Fenster geöffnet, und in der unteren rechten Bildschirmecke befindet sich das Zeichenfenster. Dort entwirft er mit der Maus gerade das Titelbild für seine (auch neue) internationale ATARI-ST-Fachzeitschrift. Hans-Otto malt munter drauflos und benutzt eifrig alle Fähigkeiten, die das Programm bietet: Strichbreiten, Farben, Füllmuster etc. Dummerweise sind jedoch alle diese Fähigkeiten nur über die Menüzeile zu erreichen, und die befindet sich weit weg am oberen Bildschirmrand. Schon nach kurzer Zeit wird sein Arm lahm, weil er ständig mit der Maus zwischen dem oberen und unteren Bildschirmrand hin- und herfahren muß; nicht lange, und die Arbeit macht ihm keinen Spaß mehr. Frustriert schaltet er den Computer aus und widmet sich stattdessen lieber einem guten Buch: "Per Anhalter durch die Galaxis" von Douglas Adams. Und das ist auch der Grund, warum wir niemals in den Genuß einer weiteren guten ST-Zeitschrift kommen werden...

Daß das nicht so sein muß, und daß es auch anders geht, wird im folgenden bewiesen. Wer an dieser Stelle aber einwendet: "Hans-Otto braucht doch bloß das Fenster nach oben unter die Menüzeile zu verschieben!", dem ist nicht mehr zu helfen, und der kann diesen Artikel getrost vergessen. Er sollte sich stattdessen lieber ebenfalls einem guten Buch zuwenden. Wie wär's mit "Das Erbe des Zauberers" von Terry Pratchett? Und wer sich für immer vom Computer-Dasein abwenden will, sollte sich Stephen R. Donaldsons "Der Spiegel ihrer Träume" und "Einer reitet durch" zuwenden... (Da hat man was fürs Leben!)

Was dem Desktop recht ist...

Nun aber zu der Lösung des Problems, die meiner Meinung nach am vernünftigsten folgendermaßen aussieht: Statt die Menüzeile grundsätzlich immer am oberen Bildschirmrand zu positionieren, wäre es doch viel einfacher, sie am oberen Rand des gerade benutzten Fensters zu installieren. Dadurch würde der mühsame und oft lästige Weg (das wird jetzt gerade besonders bei Großmonitoren aktuell!) bis zum oberen Bildschirmrand entfallen. Außerdem könnte man unterschiedliche Fenster (Text-, Grafikfenster etc.) auch mit unterschiedlichen Menüleisten versehen, von denen aber natürlich immer nur die des gerade aktivierten Fensters auch zugänglich ist.

...ist dem Fenster billig

Genau dieses Prinzip soll anhand eines Beispielprogramms hier näher erklärt werden. Das Programm öffnet zwei Fenster mit je einer Menüzeile; mit den Fenstern kann beliebig herumgespielt werden, also: Vergrößern, Verkleinern, Verschieben etc. funktioniert so, wie es funktionieren sollte. Zusätzlich kann aber jetzt noch im jeweils aktiven Fenster auf die Menüleiste zugegriffen werden; der Einfachheit halber befindet sich in beiden Fenstern die gleiche Menüleiste. Verlassen wird das Programm durch Anklicken des Schließfeldes eines der Fenster. Eine Hardcopy des fertigen Programms zeigt Bild 1.

Bevor sich nun aber irgendjemand aber an das Abtippen des Listings heranwagt, muß erst noch mit einem RCS das zugehörige Resource-File namens "MENUDMO2.RSC" erstellt werden, das folgende Dinge enthalten muß:

1. eine Menüzeile, die den Namen MENU bekommt. Die Anzahl der

Menüs ist gleichgültig, ebenso deren Inhalt. Der erste Eintrag im Desk-Menü erhält den Namen ABOUT. Die fertige Menüzeile könnte beispielsweise so aussehen wie in Bild 2; ich gehe in den folgenden Erklärungen von diesem oder einem ähnlichen Aussehen und dem Vorhandensein von zwei Menüs aus.

2. eine Dialogbox mit der Information über das Programm; sie bekommt den Namen DABOUT, der Exitbutton den Namen OK. Im Beispiel wird die Box aus Bild 3 verwendet.

Wer an dieser Stelle schon ein bißchen mit dem fertigen Programm herumgespielt hat und nun auf das Funktionsprinzip neugierig geworden ist, sollte sich vielleicht noch einmal vor dem Weiterlesen den bereits oben erwähnten Artikel "Submenüs unter GEM" in [1] zu Gemüte führen. Dort wurden alle Grundlagen, die hier benötigt werden, bereits ausführlich behandelt. Alle Leser, denen die Submenüs zu kompliziert waren, kann ich aber beruhigen: Diesmal wird's nicht ganz so schlimm...

Aus diesem Grund möchte ich hier auch nur das Wichtigste noch einmal wiederholen: den Aufbau einer Menüleiste aus der Sicht des Betriebssystems. Eine ähnliche Zeichnung wie Bild 4, die den Menübaum der Menüleiste aus Bild 2 darstellt, war zwar ebenfalls schon in besagtem Artikel zu finden, der Ansatzpunkt für das Verlegen der Menüzeile ist jedoch an einer ganz anderen Stelle zu suchen als der Ansatzpunkt für die Submenüs.

Während man sich für die Submenüs mittels benutzerdefinierter Objekte in das Zeichnen der Menüs "einklinken" mußte, ist dieses umständliche Vorgehen zum Verlegen der Menüleiste in ein Fenster oder an eine andere (beliebige!) Stelle des Bildschirms nicht notwendig.

Der Dreh- und Angelpunkt des ganzen Verfahrens ist deshalb auch nicht - im Gegensatz zu den Submenüs - in den Blättern des Menübaumes zu suchen, sondern genau am entgegengesetzten Ende, nämlich der Wurzel.

Das Wurzelobjekt mit der Bezeichnung IBOX (= Invisible Box) und der Indexnummer 0 ist dabei ein unsichtbarer Rahmen, der den Rest des gesamten Baumes umspannt und standardmäßig Bildschirmgröße besitzt. Wie gesagt, diese Box enthält den gesamten Rest des Baumes, oder mit anderen Worten: Die Koordinaten sämtlicher Kind-Objekte, die ja alle letztendlich ihren Ursprung im Wurzelobjekt haben, beziehen sich - wie in Objektbäumen üblich - relativ auf dessen Position.

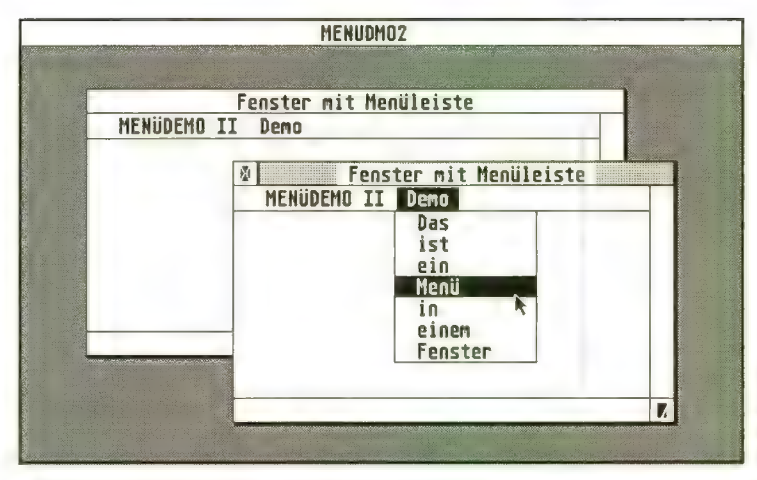


Bild 1: Das lauffähige Programm sollte sich etwa so demonstrieren.

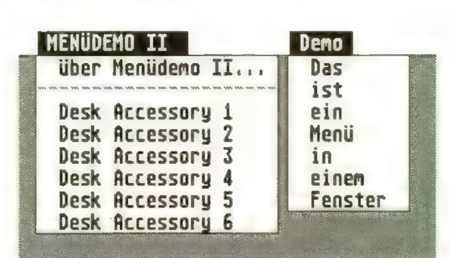


Bild 2: Die im Beispiel verwendete Menüzeile.

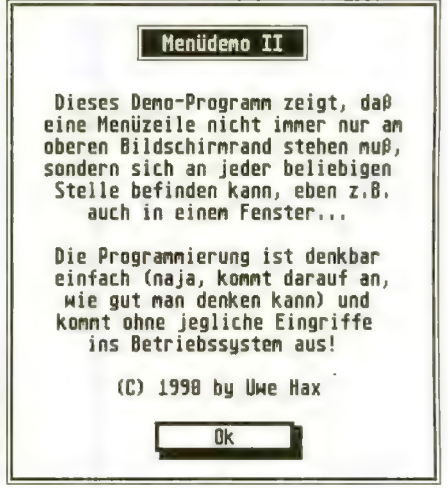


Bild 3: Die im Beispiel verwendete Info-Box.

Schein...

Werden die Koordinaten dieser Box nun geändert, erfolgt das Zeichnen der Menüzeile bei der Anmeldung mit menu_bar() prompt nicht mehr am oberen Bildschirmrand, sondern an der den Koordinaten entsprechenden Bildschirmposition. Wenn man dies einfach mal ausprobiert, wird man sehr schnell feststellen, daß das gewünschte Ergebnis damit scheinbar (bis auf eine Kleinigkeit) schon erreicht ist und die Menüzeile rein optisch jetzt an einer anderen Stelle liegt, als man es gewohnt ist.

Bei der eben erwähnten Kleinigkeit handelt es sich darum, daß das Ende der Menüzeile aufgrund der veränderten Position jetzt plötzlich über das Bildschirmende hinaus- und zum linken Bildschirmrand wieder hineinragt. Das ist jedoch auch ziemlich einfach zu ändern: Das Objekt mit der Indexnummer 1 stellt nämlich den weißen Streifen der Menüzeile dar; also muß man hier nur noch die neue Breite eintragen, und auch dieses Problem ist erledigt.

...und Wirklichkeit

Hurra, Ziel erreicht! Könnte man meinen. - Aber: Denkste! Zeichnet man eine Menüleiste mal zu weit rechts, so daß auch einige der Menütitel nicht mehr auf den Bildschirm passen, wird man nach kurzer Zeit frustriert feststellen, daß sich das Betriebssystem durch nichts, aber auch wirklich nichts davon abhalten läßt, diese überzähligen Menütitel am linken Bildschirmrand einzublenden. Das gleiche Problem tritt ebenfalls auf, wenn man eine lange Menüleiste in einem schmalen Fenster unterbringen möchte; die Menütitel, die nicht mehr in das Fenster hineinpassen, werden rechts über den Rand des Fensters hinaus gezeichnet, ohne daß man's verhindern könnte.

Wenn man nun aber die Absicht hat, die Menüzeile in ein Fenster zu verlegen, ist es unabdingbar, daß solche Effekte nicht auftreten. Denn bekannterweise kann man Fenster (normalerweise) beliebig vergrößern, verkleinern und außerdem auch noch verschieben, und das auch weit über den rechten Bildschirmrand hinaus! Beim Zeichnen der Menüzeile werden nun aber dummerweise weder irgendwelche Clipping-Grenzen noch irgendwelche ähnlichen Manipulationen an der Struktur der Menüzeile selbst beachtet. "Warum auch?", sollte man meinen; am oberen Bildschirmrand ist so etwas ja schließlich auch nicht nötig. Traurig, aber wahr!

Bei diesem Problem angelangt, dürften die meisten "Hobby-Hacker" das Problem vermutlich Problem sein lassen und sich

doch lieber dem oben bereits erwähnten guten Buch zuwenden. Aber das kann man ja schließlich auch verstehen: warum sich mit etwas herumschlagen, was GEM ja angeblich doch nicht kann?

An dieser Stelle könnte nun eventuell jemand auf den Gedanken kommen: Warum sorgen wir nicht einfach dafür, daß das Fenster nicht schmaler werden kann als die Menüzeile breit ist? Dann hätten wir auch die oben beschriebenen Probleme nicht mehr! Der Gedanke hat zwar etwas für sich, ist jedoch in einem vernünftigen Programm kaum akzeptabel, denn die wenigsten Programme besitzen eine Menüzeile mit nur zwei oder drei Menüs. Sobald die Menüzeile etwas länger ist, wäre die Fensterbreite immer irgendwo in der Nähe der Bildschirmbreite anzusiedeln; und wer hat schon Lust, mit so einem Programm zu arbeiten? Genauso gut könnte man dann auf Fenster auch völlig verzichten!

Beim Zeichnen der Menüleiste hält sich das Betriebssystem nur an die Informationen, die es im linken Teil des Menübaums (hier die Objekte 1 bis 4) vorfindet, d.h. die Menüleiste wird genau so hoch und so breit gezeichnet, wie es die BOX mit der Indexnummer 1 angibt.

geln deren Lage und Größe wider. Und genau diese Titel sind es, die in jedem Falle gezeichnet werden und uns deshalb maßgeblich bei unseren Vorhaben stören.

Die einzigen beiden Möglichkeiten, die ich gefunden habe, um dieses Zeichnen zu verhindern, sind folgende: Zum einen

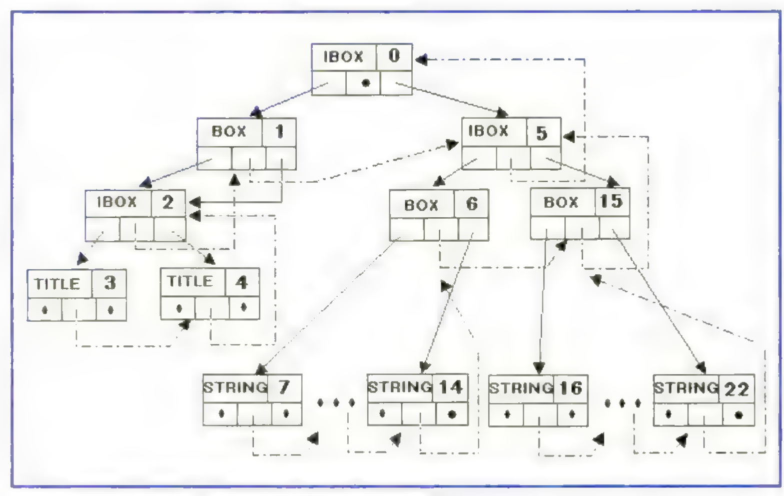


Bild 4: Der Objektbaum der Menüleiste aus Bild 1

Hokuspokus, verschwindibus!

Was nun tun? Sehen wir uns noch einmal die schematische Darstellung des Menübaums (Bild 4) an: die Zusammenhänge zwischen Sein und Nicht-Sein (die Menüleiste, so wie sie sein soll, und die Menüleiste, so wie sie nicht sein soll), mit anderen Worten: der Unsinn auf dem Bildschirm, werden dort offensichtlich.

Die IBOX mit der Indexnummer 2 dagegen gibt an, wann sich die Menüs überhaupt angesprochen zu fühlen haben, d.h. dies ist das Rechteck, in dem sich der Mauszeiger befinden muß, damit das Betriebssystem auf ihn reagiert und das entsprechende Menü ausklappt. Die restlichen Objekte (in diesem Fall nur zwei) stellen die Titel der Menüs dar und spie-

kann man dem Betriebssystem vorgaukeln, daß es über weniger Menüs verfügt, als dies tatsächlich der Fall ist, indem man die entsprechenden Zeiger umsetzt.

Zum anderen gibt es jedoch auch noch eine wesentlich elegantere Methode, die zum Glück auch beim Zeichnen der Menüzeile funktioniert: Man setzt ganz einfach in allen Objekten, sprich: den Menü-

```
 1:  /**************************************************/
 2:  /* Menüdemo II V3.02                              */
 3:  /* - Jedem Fenster seine Menüleiste -             */
 4:  /* 1990 Uwe Hax / (c) MAXON Computer GmbH 1990    */
 5:  /**************************************************/
 6:
 7:  #include <gemdefs.h>
 8:  #include <obdefs.h>
 9:  #include <osbind.h>
10:  #include <portab.h>
11:
12:  #include "menudmo2.h"
13:
14:  #define DESKTOP 0
15:
16:  WORD contrl[12];
17:  WORD intin[128];
18:  WORD ptsin[128];
19:  WORD intout[128];
20:  WORD ptsout[128];
21:  WORD int_in[11];
22:  WORD int_out[57];
23:
24:  WORD dummy;
25:
26:  FDB memory,screen;  /* zum Sichern der Menüzeile
                            als Grafik */
27:  WORD vdi_handle;    /* Handle für VDI-Routinen */
28:  WORD w1_handle,w2_handle;  /* Fensterkennungen */
29:  LONG adr1,adr2;     /* Speicheradr. für Grafik */
30:
31:  main()
32:  {
33:    OBJECT *menu;     /* Zeiger auf den Menübaum */
34:    OBJECT *about;    /* Zeiger auf die Info-Box */
35:    REG WORD mesg_buff[8];      /* Message-Buffer */
36:    WORD x_desk,y_desk,w_desk,h_desk;  /* Koordi-
                  naten und Größe des Desktop */
37:    REG WORD ende=FALSE;          /* Programmende */
38:    WORD x,y,w,h;   /* Koordinaten für Dialogbox */
39:
40:    /* Programm anmelden */
41:    appl_init();
42:    vdi_handle=graf_handle(&dummy,&dummy,&dummy,
                              &dummy);
43:    open_vwork();
44:    hide_mouse();
45:
46:    /* Resource-File mit Menüzeile und Info-Box
          laden */
47:    if (!rsrc_load("menudmo2.rsc"))
48:    {
49:      form_alert(1,"[3][MENUDMO2.RSC nicht
                   gefunden!][ Abbruch ]");
50:      appl_exit();
51:      exit();
52:    }
53:    rsrc_gaddr(R_TREE,MENU,&menu);
54:    rsrc_gaddr(R_TREE,DABOUT,&about);
55:
56:    /* Fenster 1 erzeugen und öffnen */
57:    wind_get(DESKTOP,WF_WORKXYWH,&x_desk,&y_desk,
                &w_desk,&h_desk);
58:    w1_handle=wind_create(NAME | CLOSER | MOVER |
                  SIZER,x_desk,y_desk,w_desk,h_desk);
60:    wind_set(w1_handle,WF_NAME," Fenster 1 mit
                Menüleiste ",0,0);
61:    wind_open(w1_handle,50,50,400,200);
              /* Fenstergröße für höchste Auflösung */
62:
```

Heaven's Gate

ATARI ST

HARLEKIN

Inside heaven: reset-fester Editor, Diskmonitor (Dateien und Sektoren), Formatierprogramm, reset-feste Datenbank und Terminplaner, Taschenrechner, reset-feste RAM-Disk, Tastatur-Macro-Programmer, Kopierprogramm, luxuriöses Diskutility, einstellbarer Wecker und (Stopp-)Uhr, intelligenter Druckerspooler mit optionaler Ausgabe auf Disk, Bildschirmschoner, Systemzeichensatzeditor und -installer, Druckkonverter, ASCII-Tabelle mit Übernahme in GEM-Programme, Terminalprogramm, Tastatur-Reset (Warm- und Kaltstart), eigene System-Fileselectorbox, stark erweitertes Kontrollfeld, Quickmouse, RS232-Einstellung,... u.v.a.m.

1 MB und Monochrommonitor erforderlich

Harlekin ab Versand inkl. Porto u. Verpackung DM 136,50

Auslandsbestellungen nur gegen Vorauskasse. Prospekt anfordern.

DM 129,-

Unverbindlich empfohlener Verkaufspreis



MAXON Computer GmbH • Schwalbacher Str. 52 • 6236 Eschborn • Tel.: 06196/481811 • Fax: 06196/41885

titeln, die nicht gezeichnet werden sollen, weil sie nicht mehr auf den Bildschirm oder ins Fenster passen, das HIDETREE-Flag. Und schon sieht das Ergebnis aus wie gewünscht: die Menüzeile besteht plötzlich scheinbar aus wesentlich weniger Menüs als vorher! Allerdings muß an dieser Stelle auch noch die Breite des Reaktions-Rechtecks (IBOX 2) für das Betriebssystem entsprechend angepaßt werden, damit alles korrekt funktioniert.

Zu erwähnen ist hier vielleicht noch, daß es völlig ausreicht, die Menütitel beim Zeichnen der Menüleiste auszublenden, um zu verhindern, daß die entsprechenden Menüs ausgeklappt werden können. Wo keine anzuwählenden Titel sind, da klappen auch keine Menüs heraus! Um ein Menü am Ausklappen zu hindern, kann man übrigens auch einfach im entsprechenden Titel das DISABLED-Flag setzen; leider reicht das (logischerweise) jedoch nicht aus, um das Zeichnen der Titel zu verhindern.

Eines hab' ich noch...

Nachdem das Funktionsprinzip der gekürzten Menüzeile geklärt ist, komme ich nun zum nächsten Problem. (Doch, doch, es gibt noch eins!)

Und zwar ist das Problem wiederum bei der fehlenden Beachtung irgendwelcher Clipping-Grenzen zu suchen. Durch diese massive Beeinträchtigung - wie schon beim Zeichnen der Menüzeile festgestellt - kommt man spätestens dann in arge Bedrängnis, wenn es um das Neuzeichnen oder auch Redraw bestimmter Teilbereiche des Bildschirms bzw. der Fenster geht.

Solange die komplette Menüzeile neu gezeichnet werden muß, ist noch alles in Ordnung; dürfen jedoch bleistiftsweise nur die rechte Hälfte oder etwa die oberen drei Pixelzeilen der Menüleiste neugezeichnet werden, weil der Rest von einem Fenster überlappt wird, steckt man wieder mal in einem Dilemma und muß sich erneut fragen: Was tun?

Malen und Zeichnen

Die Lösung ist an dieser Stelle jedoch wesentlich unkomplizierter als im obigen Fall; man muß nur genug Phantasie haben, Routinen auch mal zweckentfremdet zu verwenden: Sobald eine Menüzeile installiert wurde, wird sie mittels der VDI-Routine vro_cpyfm() vom Bildschirm an einen reservierten Speicherplatz kopiert und steht anschließend fürs Redraw jederzeit zur Verfügung. Im Gegensatz zur AES-Routine menu_bar() hält sich die VDI-Routine vro_cpyfm() beim Zurückkopieren auf den Bildschirm nämlich durchaus an etwaige Clipping-Grenzen...

Dieses Vorgehen funktioniert aber natürlich nur dann einwandfrei, wenn man die Menüzeile nach jeder Änderung (zum Beispiel durch die Größenänderung eines Fensters) erneut als Grafik abspeichert, damit beim Redraw nicht plötzlich etwas völlig anderes an der entsprechenden Stelle steht. Man kann jetzt also statt menu_bar() plötzlich vro_cpyfm() zum (Neu-)Zeichnen der Menüzeile verwenden, solange sich Position oder Aussehen der verwendeten Menüzeile nicht ändern. In diesem Fall muß die Menüleiste dann wieder erneut mit menu_bar() angemeldet und als Grafik abgespeichert werden. - Mit anderen Worten: Es ist plötzlich ein völlig normales Redraw möglich!

Zur Funktionsweise von vro_cpyfm() möchte ich erneut auf [1] verweisen: die VDI-Routine wurde dort bereits ausführlich beschrieben.

Es läuft!

Nachdem nun alle Probleme und Klarheiten restlos beseitigt sind, sehen wir uns noch kurz den Aufbau und Ablauf des Demoprogramms an.

Im Hauptteil main() werden nach Anmeldung des Programms und Einladen des Resource-Files zwei Fenster geöffnet und in beiden die Menüzeile installiert. Die Größe der Fenster ist dabei auf die höchste Auflösung ausgelegt; wer einen Farbmonitor benutzt, sollte die Fenstergröße entsprechend anpassen. Anschließend läuft das Programm solange in einer Schleife, bis das Schließfeld eines der beiden Fenster angeklickt wird.

In der Zwischenzeit werden alle ankommenden Nachrichten, die die Fensterverwaltung betreffen, ausgewertet und entsprechende Reaktionen veranlaßt. Dabei reagiert das Programm beispielhaft auf alle an einem Fenster möglichen Manipu-

```
63:     /* Menüzeile in Fenster 1 installieren */
64:     install_menu(w1_handle,menu,0,0,0,0);
65:
66:     /* Fenster 1 mit Inhalt füllen */
67:     fill_window(w1_handle,menu);
68:
69:     /* Fenster 2 erzeugen und öffnen */
70:     w2_handle=wind_create(NAME | CLOSER | MOVER |
71:               SIZER,x_desk,y_desk,w_desk,h_desk);
72:     wind_set(w2_handle,WF_NAME," Fenster 2 mit
              Menüleiste ",0,0);
73:     wind_open(w2_handle,100,100,500,250);
             /* FenstergröPe für höchste Auflösung */
74:
75:     /* Menüzeile in Fenster 2 installieren */
76:     install_menu(w2_handle,menu,0,0,0,0);
77:
78:     /* Fenster 2 mit Inhalt füllen */
79:     fill_window(w2_handle,menu);
80:
81:     graf_mouse(ARROW,&dummy);
82:     show_mouse();
83:
84:     /* Nachrichten verarbeiten, bis Programm
           beendet wird */
85:     do
86:     {
87:       evnt_mesag(mesg_buff);
88:
89:       hide_mouse();
90:       wind_update(BEG_UPDATE);
91:
92:       switch (mesg_buff[0])
93:       {
94:         case WM_REDRAW:    /* Fenster neuzeichnen */
```

```
95:                 redraw(mesg_buff[3],menu,mesg_buff[4],
96:                        mesg_buff[5],mesg_buff[6],
                           mesg_buff[7]);
97:                        break;
98:
99:           case WM_MOVED:      /* Fenster auf neue
                                     Koordinaten setzen */
100:                wind_set(mesg_buff[3],WF_CURRXYWH,
101:                         mesg_buff[4],mesg_buff[5],
                             mesg_buff[6],mesg_buff[7]);
102:
103:                /* ACHTUNG: kein BREAK!! */
104:                /* die folgenden Routinen müssen auch
105:                   noch ausgeführt werden: */
106:
107:          case WM_SIZED:     /* Menüzeile anpassen
                                    und... */
108:                install_menu(mesg_buff[3],menu,
                                mesg_buff[4],
109:                 mesg_buff[5],mesg_buff[6],
                                mesg_buff[7]);
110:
111:                /* ...Fenster neu ausfüllen */
112:                fill_window(mesg_buff[3],menu);
113:
114:                       break;
115:
116:          case WM_TOPPED:    /* Fenster in den
                                    Vordergrund bringen */
117:                wind_set(mesg_buff[3],WF_TOP,0,0,0,0);
118:
119:                    /* und Menüzeile installieren */
120:              install_menu(mesg_buff[3],menu,0,0,0,0);
121:                       break;
122:
```

→

lationen; lediglich auf die Scroll-Balken habe ich verzichtet, da sie in keinster Weise die Darstellung der Menüzeile im Fenster beeinflussen.

Im einzelnen werden dabei folgende Nachrichten ausgewertet:

- WM_REDRAW: Diese Nachricht tritt beispielsweise auf, wenn sich Fenster überlappen und Teile eines Fensters neu gezeichnet werden müssen. Die neu zu zeichnenden Rechtecke müssen dann aus der Rechteckliste ausgelesen und neu gezeichnet werden.

- WM_MOVED: Nach Verschieben eines Fensters muß das Fenster an den neuen Koordinaten gezeichnet werden; anschließend werden die Menüzeile an der neuen Position installiert und der Fensterinhalt neu gemalt.

- WM_SIZED: Hier sind die gleichen Aktionen wie bei WM_MOVED notwendig; nur das Neuzeichnen des Fensters an den neuen Koordinaten entfällt - diese bleiben natürlich gleich. Zu beachten ist, daß auch hier die Menüzeile neu installiert werden muß, weil sich aufgrund der neuen Fenstergröße die Anzahl der darzustellenden Menüs geändert haben kann.

- WM_TOPPED: Nachdem das entsprechende Fenster in den Vordergrund gebracht wurde, muß auch hier erneut die Menüzeile installiert werden, da sie natürlich bisher in dem jetzt im Hintergrund befindlichen Fenster installiert war.

- WM_CLOSED: Hier wird lediglich ein Flag gesetzt, um anschließend die Schleife beenden zu können.

- MN_SELECTED: Wurde der ABOUT-Eintrag im Desk-Menü angeklickt, wird hier die entsprechende Mitteilung ausgegeben.

Wurde die Schleife beendet, werden die Fenster geschlossen und gelöscht, anschließend noch die Menüzeile und benötigter Speicher wieder freigegeben und dann das Programm verlassen.

Installationsarbeiten

Die Routine install_menu() installiert - wie der Name schon sagt - die Menüleiste innerhalb eines Fensters. Dazu müssen folgende Parameter übergeben werden:

w_handle:	die Fenster-Kennung, die man bei der Erzeugung eines Fensters mit wind_create() erhält.
menu:	die Adresse der Menüzeile, die man mittels rsrc_gaddr() erhält.
x,y:	die Koordinaten der neuen Fensterposition
width,height:	die neue Größe des Fensters

Enthalten die letzten vier Variablen alle den Wert Null, werden die augenblickliche Position und Größe beibehalten, indem diese Variablen innerhalb der Funktion initialisiert werden. Außerdem wird dafür gesorgt, daß das Fenster eine gewisse Minimalgröße nicht unterschreitet; die Werte können eigenen Bedürfnissen angepaßt werden.

Anschließend wird die Menüzeile wie oben beschrieben durch Ändern der Koordinaten des Wurzelobjekts auf die neue Position gesetzt und ihre Breite an das Fenster angepaßt.

Daraufhin wird in einer Schleife überprüft, welche Menüs noch in das Fenster bzw. auf den Bildschirm passen, und alle Menüs, die diese Bedingung nicht erfüllen, werden durch Setzen des HIDETREE-Flags abgeschaltet.

Wichtig ist an dieser Stelle, daß sowohl geprüft wird, ob der Menütitel noch in das Fenster, als auch, ob das Menü selbst in der Breite noch paßt. Klappt ein Menü über die untere Begrenzung eines Fensters hinaus, hat das keinen Einfluß auf die Darstellung am Bildschirm; anders sieht es jedoch aus, wenn ein Menü über die rechte Grenze eines Fensters hinausragt. In diesem Fall macht sich dummerweise ein kleiner Fehler des Betriebssystems bemerkbar, der normalerweise nicht auffällt: Wird ein Menü wieder eingeklappt, bleibt nämlich der obere Rand des vorher ausgeklappten Menüs auf dem Bildschirm stehen und wird nicht wieder gelöscht. Im Normalfall, d.h. wenn sich die Menüleiste am oberen Bildschirmrand befindet, ist

```
123:         case WM_CLOSED:     /* Programmende */
124:                             ende=TRUE;
125:                             break;
126:
127:         case MN_SELECTED:   /* Menüeintrag wurde
                                    angewählt */
128:             if (mesg_buff[4]==ABOUT)
129:             {
130:             /* Info-Box zeichnen und beenden */
131:             about[OK].ob_state &=~SELECTED;
132:             form_center(about,&x,&y,&w,&h);
133:             form_dial(FMD_START,x,y,w,h,x,y,w,h);
134:             objc_draw(about,ROOT,MAX_DEPTH,
                           x,y,w,h);
135:             show_mouse();
136:             form_do(about,0);
137:             hide_mouse();
138:             form_dial(FMD_FINISH,x,y,w,h,x,y,w,h);
139:             }
140:             /* Menütitel wieder zurücksetzen */
141:             menu_tnormal(menu,mesg_buff[3],TRUE);
142:                             break;
143:       }
144:
145:       wind_update(END_UPDATE);
146:       show_mouse();
147:     }
148:     while (!ende);
149:
150:     /* Fenster schließen und löschen */
151:     wind_close(w1_handle);
152:     wind_close(w2_handle);
153:     wind_delete(w1_handle);
```

```
154:     wind_delete(w2_handle);
155:
156:     /* Menüzeile entfernen und Prgm verlassen */
157:     menu_bar(menu,FALSE);
158:     Mfree(adr1);
159:     Mfree(adr2);
160:     v_clsvwk(vdi_handle);
161:     appl_exit();
162:   }
163:
164:
165:   install_menu(w_handle,menu,x,y,width,height)
166:   REG WORD w_handle;      /*Fenster-Kennung */
167:   REG OBJECT *menu;       /*Adresse Menüleiste */
168:   WORD x,y,width,height; /*neue GröPe des Fensters;
                               enthalten
169:                     alle Werte 0, wird die augenblick-
170:                     liche Größe&Posit. beibehalten */
171:   {
172:     WORD wx,wy,ww,wh;  /* Koord. Arbeitsbereich */
173:     WORD cx,cy,cw,ch;  /* Koord. Fenster-Gesamt */
174:     WORD x_desk,y_desk,w_desk,h_desk;
                          /* Bildschirmausmaße     */
175:     REG WORD index1,index2;  /* Menü-Indices    */
176:     REG WORD array[8]; /* Eingabefeld, div.Rou. */
177:     WORD m_width;      /* Menü-Breite          */
178:     WORD resolution;   /* Auflösung            */
179:     WORD rows,cols;    /* Bildschirmgröße      */
180:     WORD tail=menu[0].ob_tail;  /* div. Abk.    */
181:     WORD tail2=menu[tail].ob_head;
182:     WORD head=menu[0].ob_head;
183:     WORD head2=menu[head].ob_head;
184:     WORD temp1,temp2;                         →
```

dieser Fehler nicht sichtbar, weil die Linie direkt unterhalb der Menüzeile deckungsgleich mit der oberen Abschlußkante der Menüs ist. In unserem Fall befindet sich diese Linie jedoch nur innerhalb des Fensters, und deshalb dürfen die Menüs auch nur an dieser Stelle aufklappen; alle Menüs, die über den inneren rechten Rand eines Fensters oder über den rechten Bildschirmrand hinausragen, dürfen demzufolge nicht geöffnet werden können.

Sind alle Änderungen durchgeführt worden, kann die Menüleiste ganz normal mit menu_bar() angemeldet werden.

Sofern install_menu() das erste Mal aufgerufen wurde, werden im folgenden einige nur einmal notwendigen Initialisierungen für vro_cpyfm() durchgeführt und anschließend als Quellkoordinaten die Koordinaten der Menüleiste eingetragen, während der zuvor angeforderte Speicher als Ziel für die Kopieraktion herhalten muß. Für genauere Informationen zu dieser Routine verweise ich erneut auf [1] und auf [2].

Leere Fenster sind auch voll

Die Funktion fill_window() benötigt als Eingabeparameter lediglich die Fensterkennung und die Adresse der Menüzeile. Mit Hilfe der Fensterkennung wird der Arbeitsbereich des entsprechenden Fensters ermittelt und dieser dann mit Weiß aufgefüllt. Außerdem wird noch die bereits erwähnte schwarze Linie unterhalb der Menüleiste gezeichnet.

Soll ein Fenster noch weiteren Inhalt haben und nicht nur "leer" sein, müssen die dazu notwendigen Ergänzungen an dieser Stelle erfolgen.

Listige Rechtecke

Die Redraw-Routine mit dem sinnigen Namen redraw() benötigt die gleichen Eingabeparameter wie die Funktion install_menu().

Nach diversen Initialisierungen, wie zum Beispiel der Ermittlung der Koordinaten und Größe des Arbeitsbereiches und der Bildschirmgröße,wird zuerst die Breite des Arbeitsbereichs für vro_cpyfm() angepaßt und das erste Rechteck aus der Rechteckliste geholt.

Anschließend wird in einer Schleife überprüft, ob sich der neu zu zeichnende Bereich mit dem aus der Rechteck-Liste geholten Rechteck überlappt. Dazu wird die Funktion rc_intersect() benutzt, die überprüft, ob sich die beiden Eingabe-Rechtecke überschneiden, und im Erfolgsfall TRUE zurückliefert. Diese Funktion ist zwar nirgends dokumentiert, ist aber nichtsdestoweniger in allen (?) Compiler-Bibliotheken vorhanden.

Überlappen sich die Rechtecke, werden der Clipping-Bereich auf den wiederherzustellenden Fensterbereich beschränkt, der Fensterinhalt und die abgespeicherte Grafik der Menüleiste neu gezeichnet und letztendlich das nächste Rechteck aus der Rechteck-Liste geholt, damit die Schleife für dieses erneut ausgeführt werden kann.

Die restlichen Routinen, open_vwork(), hide_mouse() und show_mouse() bedürfen wohl keiner Erklärung.

Nachschlag

Zum Schluß möchte ich noch einige Bemerkungen zu den bereits mehrfach erwähnten Submenüs aus [1] loswerden, die sich erst nach Abdruck des Artikels ergeben haben.

Zum einen habe ich dort behauptet, daß das abgedruckte Demoprogramm ohne Änderung nicht einwandfrei funktioniert, wenn vorher mit GDOS gebootet wurde. Diese Aussage muß ich zurücknehmen bzw. relativieren. Mir stand zum Test nur eine ziemlich alte Version von GDOS zur Verfügung, und für diese stimmte die obige Behauptung tatsächlich; in Verbindung mit neueren GDOS-Versionen - wie zum Beispiel AMCGDOS 3.20 - läuft das Demoprogramm jedoch auch ohne jegliche Änderung einwandfrei.

Zum anderen habe ich an dieser Stelle noch eine kleine Verbesserung zu den Submenüs anzubieten; und zwar betrifft diese Verbesserung das Einklappen der Menüs. Bisher gab es das Problem, daß in den meisten Fällen beim Anklicken eines Submenü-Eintrags zwar das Submenü vom Bildschirm verschwand, das Menü selber aber vom Betriebssystem erst dann wieder eingeklappt wurde, wenn erneut ein Klick irgendwohin erfolgte. Insbesondere gab es dann große Schwierigkeiten, wenn Teile des Bildschirms neu gezeich-

```
185:
186:    static WORD first=TRUE;     /* erster Aufruf? */
187:
188:    /* GröPe des Arbeitsbereiches und GesamtgröPe
           des Fensters ermitteln */
189:    wind_get(w_handle,WF_WORKXYWH,&wx,&wy,&ww,&wh);
190:    wind_get(w_handle,WF_CURRXYWH,&cx,&cy,&cw,&ch);
191:
192:    /* sind alle übergebenen Werte Null, wird die
           augenblickliche Größe und
193:       Position beibehalten */
194:    if (!x && !y && !width && !height)
195:    {
196:      x=cx;
197:      y=cy;
198:      width=cw;
199:      height=ch;
200:    }
201:
202:    /* Minimalhöhe und -breite: willkürlich
           gewählt */
203:    if (height<60)
204:      height=60;
205:    if (width<100)
206:      width=100;
207:
208:    /* neue Fenstergröße setzen */
209:    wind_set(w_handle,WF_CURRXYWH,x,y,width,
                 height);
210:
211:    /* neue Position der Menüzeile */
212:    menu[0].ob_x=x+1;
213:    menu[0].ob_y=y+(ch-wh)/2;     /* "/2" entfällt,
```

```
                               falls das Fenster keine
214:                           untere Leiste enthält */
215:
216:    /* geht die Menüzeile über das Bildschirmende
           hinaus? */
217:    wind_get(DESKTOP,WF_WORKXYWH,&x_desk,&y_desk,
                 &w_desk,&h_desk);
218:    if (menu[0].ob_x+width-cw+ww>x_desk+w_desk)
219:      width=x_desk+w_desk-menu[0].ob_x+cw-ww-1;
                  /* Breite nur bis Bildschirmende */
220:
221:    /* Breite der Menüzeile = Breite des Fensters
           bzw. bis Bildschirmende */
222:    menu[head].ob_width=width-cw+ww;
223:
224:    index1=menu[head2].ob_head;
225:    index2=tail2;
226:    menu[head2].ob_width=0 ;
227:    while (menu[index1].ob_type==G_TITLE)
228:    {
229:      /* Breite des Menütitels */
230:      temp1=menu[0].ob_x+menu[head].ob_x+
                menu[head2].ob_x+menu[index1].ob_x+
231:            menu[index1].ob_width;
232:
233:      /* Breite des Menüs */
234:      temp2=menu[0].ob_x+menu[tail].ob_x+
              menu[index2].ob_x+menu[index2].ob_width;
235:
236:      /* welche Menüs passen noch auf den
             Bildschirm bzw. ins Fenster? */
237:      if ((temp1>=x_desk+w_desk) || (temp2>=x_desk+
                w_desk) || →
```

net werden sollten und das Menü noch ausgeklappt war. Ich weiß zwar immer noch nicht, warum das Betriebssystem die Menüs manchmal einklappt und meistens nicht, aber ich habe eine Lösung gefunden, um dieses Problem zu umgehen.

Sobald die Variable subnum einen Rückgabewert enthält und das entsprechende Menü noch ausgeklappt ist, wird dem Betriebssystem mittels appl_tplay() einfach ein Mausklick vorgegaukelt und damit das Menü eingeklappt. Da es sich bei appl_tplay() um eine AES-Routine handelt, darf sie nicht innerhalb der Submenü-Routinen durchgeführt werden, und der Anwender muß sich deshalb leider selbst darum kümmern.

Folgende Änderungen muß man dazu an MENUDEMO.C durchführen :

- Zuerst wird der Mausklick für appl_tplay() definiert. Zu beachten ist dabei, daß es sich eigentlich um zwei Ereignisse handelt, nämlich "Maustaste drücken" und "Maustaste loslassen". Genaue Informationen zum Aufbau einer solchen Nachricht sind in jedem einigermaßen brauchbaren Buch zu finden, beispielsweise in [3]. Folgende Zeile ist nach Zeile 25 einzufügen:

```
WORD click[8]={ 0,1,1,1,0,1,0,0 };
```

- Nach Zeile 115 wird der Mausklick dann ausgeführt, sofern noch ein Menü ausgeklappt ist. Dabei wird hier natürlich nur auf zwei Menüs getestet, denn mehr hatte das Demoprogramm bekanntlich nicht. Hier die entsprechenden Zeilen:

```
if ((menu[DATEI].ob_state & SELECTED) ||
    (menu[FONT].ob_state & SELECTED))
  appl_tplay(click,2,100);    /* Mausclick =>
                              Menü klappt ein */
```

Listing: Hier wird auf 2 Menüs abgefragt.

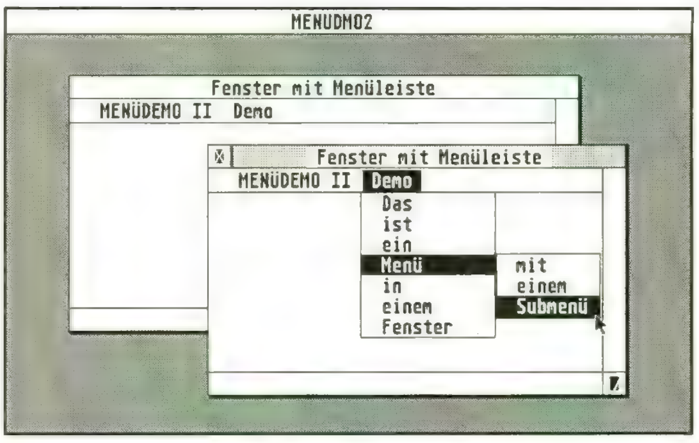


Bild 5: Auch Menüleisten in Fenstern kann man mit zusätzlichen Submenüs versehen.

Zu guter Letzt

Am Ende angelangt, möchte ich nun noch darauf hinweisen, daß man die ganze Sache mit den Menüleisten und den Submenüs natürlich auch noch auf die Spitze treiben kann, indem man nicht nur die Menüleiste in ein Fenster verlegt, sondern in dieselbige dann auch noch Submenüs installiert... (verblüffte Gesichter garantiert!) Daß dies ohne größeren Aufwand machbar ist, zeigt schlußendlich noch Bild 5. Die Änderungen der Submenü-Routinen halten sich dabei in Grenzen und sollten eigentlich von jedem einigermaßen begnadeten Programmierer in kurzer Zeit durchzuführen sein...

Wer an dieser Stelle angelangt immer noch nicht genug von Menüs und Menüleisten hat, den kann ich ja schon mal darauf hinweisen, daß es möglicherweise irgendwann (wenn ich mal wieder nichts Besseres zu tun habe) noch einen dritten Teil von "Unmöglichkeiten" geben wird - mir schweben jedenfalls noch so ein paar Sachen vor, die man auch noch mit Menüs anstellen könnte...

Uwe Hax

Literatur:

[1] ST-Computer 12/89 und 1/90: Und es geht doch! - Submenüs unter GEM
[2] H. Lemcke, V. Dittmar, M. Sommer: Programmierlexikon für den ATARI ST, Hüthig Verlag
[3] H.-D. Jankowski, D. Rabich, Julian F. Reschke: ATARI ST Profibuch, Sybex Verlag

```
238:          (temp1>=x+width-cw+ww) || (temp2>=x+
                                width-cw+ww))
239:      {
240:        /* alle nicht mehr auf den Bildschirm
              passenden Menüs abschalten */
241:        while (menu[index1].ob_type==G_TITLE)
242:        {
243:          menu[index1].ob_flags=HIDETREE;
244:          index1=menu[index1].ob_next;
245:        }
246:        break;
247:      }
248:      else
249:      {
250:        /* Breite der Menüzeile anpassen */
251:        menu[head2].ob_width=menu[index1].ob_x+
                          menu[index1].ob_width;
252:
253:        /* Menü eingeschaltet */
254:        menu[index1].ob_flags=NONE;
255:
256:        index1=menu[index1].ob_next;
257:        index2=menu[index2].ob_next;
258:      }
259:    }
260:
261:    /* modifizierte Menüzeile installieren */
262:    menu_bar(menu,TRUE);
263:
264:    if (first)    /* erster Aufruf dieser
                         Funktion? */
265:    {
266:      /* Breite des Bildschirms = maximale Länge
            der Menüzeile */
267:      vq_chcells(vdi_handle,&rows,&cols);
268:
269:      adr1=Malloc((LONG)(++cols*
               (menu[head].ob_height+2))); /*
                maximale Größe  */
270:      adr2=Malloc((LONG)(cols*
               (menu[head].ob_height+2)));
271:
272:      /* sonstige Initialis. für vro_cpyfm */
273:      resolution=Getrez();
274:      screen.fd_addr=0L;
275:      memory.fd_stand=0;
276:      memory.fd_nplanes=resolution ? 2/resolution
                            : 4;
277:
278:      first=FALSE;
279:    }
280:
281:    /* Menüzeile als Grafik für Redraw sichern: */
282:    /* Quell-Koordinaten */
283:    array[0]=menu[0].ob_x+menu[head].ob_x;
284:    array[1]=menu[0].ob_y+menu[head].ob_y;
```
→

```
285:    array[2]=array[0]+menu[head].ob_width-1;
286:    array[3]=array[1]+menu[head].ob_height-1;
287:
288:    /* Zielkoordinaten */
289:    array[4]=0;
290:    array[5]=0;
291:    array[6]=menu[head].ob_width-1;
292:    array[7]=menu[head].ob_height-1;
293:
294:    /* sonstige jedesmal notwendige
           Initialisierungen */
295:    memory.fd_addr=(w_handle==w1_handle) ? adr1 :
                    adr2;
296:    memory.fd_wdwidth=array[6]/16+1;
297:    vro_cpyfm(vdi_handle,3,array,&screen,&memory);
298:  }
299:
300:
301:  fill_window(w_handle,menu)
302:  REG WORD w_handle;      /* Fenster-Kennung */
303:  OBJECT *menu;           /* Adresse Menüzeile */
304:  {
305:    REG WORD array[4];    /* Eingabefeld für
                               diverse Routinen */
306:    WORD wx,wy,ww,wh;     /* Koordinaten des
                               Arbeitsbereichs */
307:    WORD head=menu[0].ob_head;  /* als Abk. */
308:
309:    /* Größe des Arbeitsbereiches des Fensters
           ermitteln */
310:    wind_get(w_handle,WF_WORKXYWH,&wx,&wy,&ww,&wh);
311:
312:    /* Fenster und Hintergrund der Menüzeile mit
           weißer Fläche füllen */
313:    vsf_color(vdi_handle,0);
314:    array[0]=wx;
315:    array[1]=wy+menu[head].ob_height+1;
316:    array[2]=wx+ww-1;
317:    array[3]=wy+wh-1;
318:    vr_recfl(vdi_handle,array);
319:
320:    /* Linie unter der Menüzeile zeichnen */
321:    array[0]=wx;
322:    array[1]=array[3]=wy+menu[head].ob_height;
323:    array[2]=wx+ww-1;
324:    vsf_color(vdi_handle,1);
325:    v_pline(vdi_handle,2,array);
326:  }
327:
328:
329:  redraw(w_handle,menu,x,y,width,height)
330:  REG WORD w_handle;    /* Fenster-Kennung */
331:  REG WORD x,y,width,height;   /* Koordinaten
                  des zu erneuernden Bereiches */
332:  OBJECT *menu;         /* Adresse der Menüzeile */
333:  {
334:    GRECT rect1,rect2;    /* Strukturen für Rechteck
                               -Koordinaten */
335:    WORD array[8];        /* Eingabefeld für diverse
                               Routinen */
336:    WORD wx,wy,ww,wh;     /* Koordinaten des
                               Arbeitsbereiches */
337:    WORD head=menu[0].ob_head;  /* als Abk. */
338:    WORD x_desk,y_desk,w_desk,h_desk;
                                   /* BildschirmgröPe */
339:
340:    rect2.g_x=x; /*Redraw-Rechteck initialisieren*/
341:    rect2.g_y=y;
342:    rect2.g_w=width;
343:    rect2.g_h=height;
344:
345:    /* Koordinaten und Größe des Arbeitsbereiches
           ermitteln */
346:    wind_get(w_handle,WF_WORKXYWH,&wx,&wy,&ww,&wh);
347:
348:    /* Bildschirmgröße ermitteln... */
349:    wind_get(DESKTOP,WF_WORKXYWH,&x_desk,&y_desk,
                 &w_desk,&h_desk);
350:
351:    /* und damit Breite für vro_cpyfm anpassen */
352:    if (wx+ww>x_desk+w_desk)
353:      ww=x_desk+w_desk-wx;
354:
355:    /* Koordinaten und Größe des ersten neu zu
           zeichnenden Rechtecks aus
356:       der Rechteck-Liste holen */
357:    wind_get(w_handle,WF_FIRSTXYWH,&rect1.g_x,
                 &rect1.g_y,&rect1.g_w,&rect1.g_h);
358:
359:    while (rect1.g_w && rect1.g_h)  /* fertig? */
360:    {
361:      /* überlappen sich beide Rechtecke? */
362:      if (rc_intersect(&rect2,&rect1))
363:      {
364:        /* Clip-Koordinaten setzen */
365:        array[0]=rect1.g_x;
366:        array[1]=rect1.g_y;
367:        array[2]=rect1.g_x+rect1.g_w-1;
368:        array[3]=rect1.g_y+rect1.g_h-1;
369:        vs_clip(vdi_handle,TRUE,array);
370:
371:        /* Fenster-Inhalt neu zeichnen */
372:        fill_window(w_handle,menu);
373:
374:        /* Grafik der Menüzeile neu zeichnen */
375:        /* Quellkoordinaten */
376:        array[0]=0;
377:        array[1]=0;
378:        array[2]=ww-1;
379:        array[3]=menu[head].ob_height-1;
380:
381:        /* Zielkoordinaten */
382:        array[4]=wx;
383:        array[5]=wy;
384:        array[6]=wx+array[2];
385:        array[7]=wy+array[3];
386:
387:        /* sonstige jedesmal notwendige
               Initialisierungen */
388:        memory.fd_addr=(w_handle==w1_handle) ?
                          adr1 : adr2;
389:        memory.fd_wdwidth=array[2]/16+1;
390:
391:        /* alle anderen Parameter sind bereits
               gesetzt */
392:        vro_cpyfm(vdi_handle,3,array,&memory,
             &screen);
393:      }
394:
395:      /* Koordinaten und Größe des nächsten
             Rechtecks aus der Rechteck-
396:         Liste holen */
397:      wind_get(w_handle,WF_NEXTXYWH,&rect1.g_x,
                   &rect1.g_y,&rect1.g_w,&rect1.g_h);
398:    }
399:  }
400:
401:
402:  open_vwork()      /* Virtuelle Workstation
                           einrichten */
403:  {
404:    REG WORD i;
405:
406:    for (i=1; i<10; i++)
407:      int_in[i]=1;
408:    int_in[10]=2;
409:    v_opnvwk(int_in,&vdi_handle,int_out);
410:  }
411:
412:
413:  hide_mouse()       /* Maus ausschalten */
414:  {
415:    graf_mouse(M_OFF,&dummy);
416:  }
417:
418:
419:  show_mouse()       /* Maus einschalten */
420:  {
421:    graf_mouse(M_ON,&dummy);
422:  }
```

Listing 1: Das Hauptprogramm...

Listing 2: ...und die dazugehörige Header-Datei

```
1:  #define MENU 0
2:  #define ABOUT 7
3:  #define DABOUT 1
4:  #define OK 13
```

STARKE ST-BÜCHER IM SEPTEMBER!

FÜR ALLE, DIE WERT AUF TADELLOSE TEXTE LEGEN.

DATA BECKERs Rechtschreibprofi ST ist das schnelle Prüfprogramm für Ihren ATARI. Ob Berichte, Protokolle, Briefe oder wissenschaftliche Arbeiten – dieses Prüfprogramm läßt Ihren Text zu einer Visitenkarte korrekter und exakter Arbeit werden. Für die Richtigkeit bürgt ein großer Name: Das Hauptwörterbuch wurde von Langenscheidt-Mitarbeitern zusammengestellt. Insgesamt verfügt das Programm über einen Wortschatz von rund 2,5 Millionen Wörtern. Außerdem haben Sie natürlich die Möglichkeit, weitere Fachausdrücke und fremdsprachliche Begriffe in einem entsprechenden Benutzerlexikon abzulegen. Sie können Ihre Texte wahlweise sichtbar auf dem Bildschirm „gegenlesen" lassen, oder das Programm erfaßt die erkannten Fehler in einem Protokoll, das nachträglich bearbeitet werden kann. Auf Wunsch überprüft der Rechtschreibprofi auch Groß- und Kleinschreibung. Korrekturvorschläge werden automatisch erfaßt. Dabei kostet ein Korrekturdurchgang nicht viel Zeit – dank des integrierten Cache-Speichers.

DATA BECKERs Rechtschreibprofi ST
DM 99,-
ISBN 3-89011-685-X



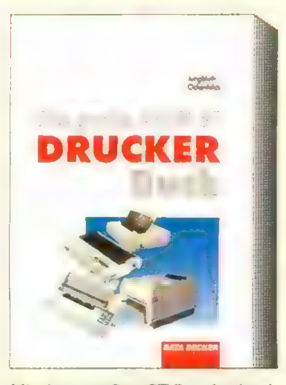
Das große Buch zu Calamus zeigt, was es heißt, DTP-Experte zu sein. So erfahren Sie z.B., wie Rahmen „numerisch" bearbeitet oder mehrere Rahmen zusammengefaßt werden, wie Rahmen für gedrehten Text entstehen, wie man Vektor- und Rastergrafiken exportiert bzw. importiert und welche Schrifteffekte es gibt. Mit vielen wichtigen Hinweisen zur Installation, zur Benutzerführung, zu Drucker und Scanner. In einem großen Praxisteil finden Sie zusätzlich noch zahlreiche Anwendungsbeispiele, die Ihnen die Leistungsfähigkeit des Programms demonstrieren und Ihnen gleichzeitig ein sicheres Gespür für eine ansprechende Gestaltung geben.

Kraus
Das große Calamus-Buch
392 Seiten, DM 39,-
ISBN 3-89011-346-X



Mit dem großen ST-Handbuch lösen Sie alle Probleme rund um den ST sozusagen im Handumdrehen. Bei Problemen mit dem Desktop genauso wie bei der Tastatur oder den Schnittstellen. Dabei werden Sie natürlich noch jede Menge interessanter Neuigkeiten über Ihren Rechner erfahren und auf eine Reihe nützlicher Tricks stoßen. Zahlreiche Tips zur Pflege Ihres Rechners und zum „Rechner-Tuning" gibt Ihnen dieses Buch genauso weiter wie viele, kleine Reparatur-Hinweise. Ein Nachschlagewerk, das Sie immer wieder einmal benutzen werden – nicht zuletzt dank seiner klaren Gliederung. Und das auch zukünftig aktuell ist: Der ST/E wird natürlich ebenfalls behandelt.

Liesert
Das große ST-Handbuch
Hardcover, 377 Seiten, DM 49,-
ISBN 3-89011-273-0



Mit dem großen ST-Druckerbuch kann eigentlich nichts mehr schiefgehen. Beginnend mit der einfachen Installation beschreibt dieser Band alles, was Sie bei der Arbeit mit Ihrem Drucker beachten müssen. Sei es die Druckersteuerung über TOS und GEM, die Druckeranpassung der verschiedenen Anwendungsprogramme oder die Arbeit mit unterschiedlichen Fonts. Daneben finden Sie einen Nachschlageteil, der Ihnen bei auftretenden Pannen weiterhilft, und eine Diskette mit vielen nützlichen Utilities (beispielsweise, um aus dem CLI heraus Batch-Dateien aufzurufen, oder ein Hardcopy-Programm).

Jungbluth/Ockenfelds
Das große ATARI-ST-Druckerbuch
Hardcover, inkl. Diskette
572 Seiten, DM 59,-
ISBN 3-89011-362-1

Signum wird gerne als das Zwischenglied zwischen Textverarbeitung und DTP-Programm bezeichnet – eine Charakterisierung, die zeigt, wie leistungsstark diese Software sein muß. Mit dem großen SIGNUM!-Buch haben Sie das Zeug, diese Textverarbeitung voll auszunutzen. Ob Diskettenoperationen, Texteinstellungen, Fußnotenverwaltung, Spaltensatz, Druckausgabe oder Bildoperationen – mit diesem Buch lernen Sie Signum/Signum II von seiner besten Seite kennen. So werden hier nicht nur sämtliche Features von Signum!, sondern auch die Möglichkeiten von Scarabus 2.0 und Headline 3.0 anhand zahlreicher praktischer Übungen beispielhaft erklärt.

Kraus
Das große SIGNUM!-Buch
346 Seiten, DM 39,-
ISBN 3-89011-313-3

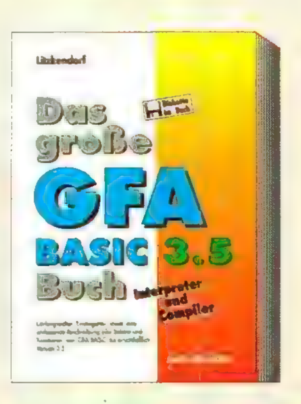
1ST WORD PLUS: JETZT GANZ EINFACH ZU BEDIENEN.



Für alle Viel- und Schönschreiber, die mit 1st Word Plus nicht nur Ihre Korrespondenz erledigen wollen: Das große Buch zu 1st Word Plus – die detaillierte Anleitung zu allen Funktionen Ihres Textprogramms. Hier werden Sie umfassend informiert. Über die vielfältigen Schrift- und Text-Layout-Möglichkeiten. Über die Grafikfähigkeit. Über die Verwaltung der Fußnoten. Mit einem ausführlichen Kapitel zur Druckeranpassung. Bei alledem wird natürlich auch die aktuelle Version 3 gebührend behandelt. Da wundert es nicht, daß Sie in diesem Band noch auf zahlreiche Tips und nützliche Tricks stoßen werden, die Ihre Arbeit mit 1st Word Plus noch effektiver machen. Denn rund die Hälfte des Bandes ist den praktischen Anwendungen gewidmet – von kleinen zu umfangreicheren Texten, von Tabellen zur Grafikeinbindung. Natürlich werden auch DTP-ähnliche Anwendungen behandelt. Das große Buch zu 1st Word Plus – mit einer Erklärung der Zusatzprogramme 1st XTRA, 1st Proportional und des Profi-Text-Moduls.

Kraus
Das große Buch zu 1st Word Plus
314 Seiten, DM 39,-
ISBN 3-89011-347-8

ALLES ÜBER GFA-BASIC!



Lassen Sie sich von einem Profi beraten, wie Sie Ihre GFA-BASIC-Programme rundherum professionell gestalten können. Ziehen Sie bei Ihrer Programmierung einfach das große GFA-BASIC-Buch zu Rate. Denn hier finden Sie alles für eine fortgeschrittene Programmierung: die praktische Anwendung der einzelnen GFA-Befehle der neuen Version 3.5, das Einbinden von System-Routinen, alles Wissenswerte zur Programmstruktur, beispielhafte Grafik- und GEM-Programmierung.

Litzkendorf
Das große GFA-BASIC-Buch
Hardc., inkl. Disk., 899 S.,
DM 59,-
ISBN 3-89011-363-X

SOFORT BESTELLEN...

...bei DATA BECKER, Merowingerstraße 30, 4000 Düsseldorf 1

Bezahlung ☐ per Nachnahme
☐ mit beiliegendem Verrechnungsscheck
(Zzgl. DM 5,- Versandkosten, unabh. von der best. Stückzahl)

Bestellung:

Vorname/Name

Straße/Nr.

PLZ/Ort

DATA BECKER

Programmer's Toolbox - Dateien

Teil 3: Einige Hilfsfunktionen

Nach diesem umfangreichen Ausflug ins Betriebssystem sollen nun einige von uns benötigte Hilfsfunktionen programmiert werden. Damit sollen Detailprobleme bei der nachfolgenden Behandlung der Kommandos möglichst ausgeklammert werden. Im Zentrum dieses Serienteils befindet sich Listing 1.3 - die Implementierung der Hilfsfunktionen.

Listing 1.3 enthält dabei Funktionen aus den drei folgenden Bereichen:

1. Verwaltung der Systemuhr: getdate putdate gettime puttime
2. Dateiverwaltung: fgetdattime fputdattime ftouch fexist fisdir confirm acp arm amv apwd acd
3. String-Behandlung: pathsplit convupper ctop ptoc onlyalpha

Die Funktionen zur Verwaltung der Systemuhr stützen sich weitgehend auf die vier GEMDOS-Funktionen zum Setzen und Auslesen der Uhr ab. Die diesbezüglichen Funktionen innerhalb von Listing 1.3 besitzen mithin lediglich die Aufgabe, dem Anwender der Systemuhr die Kodierung von Systemdatum und -zeit im GEMDOS-Zeitformat abzunehmen. Zwischen den GEMDOS-Funktionen und den Funktionen aus Listing 1.3 besteht dabei folgender funktionaler Zusammenhang:

List.1.3	GEMDOS	Funktion
getdate (Zeilen 40-49)	Tgetdate	Abfrage des Systemdatums
putdate (Zeilen 51-61)	Tsetdate	Setzen des Systemdatums
gettime (Zeilen 63-72)	Tgettime	Abfrage der Systemzeit
puttime (Zeilen 74-84)	Tsettime	Setzen der Systemzeit

Tabelle 1: Der funktionelle Zusammenhang von Listing 1.3

Die Parameter von getdate, putdate, gettime und puttime sind entsprechend der Aufgabenstellung allesamt vom Typ short bzw. vom Typ Zeiger auf short.

Etwas umfangreicher ist die Aufgabenstellung der insgesamt elf Funktionen des zweiten Bereichs von Listing 1.3. Hier sind sämtliche Funktionen zusammengefaßt, die den Umgang mit Dateien erleichtern. Obwohl GEMDOS bereits eine ganze Reihe von Funktionen zum Umgang mit Dateien bereitstellt, fehlen doch noch relativ häufig benötigte Funktionen wie etwa das Kopieren oder das Bewegen einzelner Dateien. Darüber hinaus besteht auch im Bereich der Dateiverwaltung die Notwendigkeit, einzelne GEMDOS-Funktionen mit einer aufgesetzten Funktionsschale zu versehen, wenn man komfortabel mit ihnen arbeiten will. Auch hier möchte ich wieder mit drei Funktionen beginnen, die im Zusammenhang mit dem GEMDOS-Zeitformat stehen: fgetdattime (Zeilen 114-138), fputdattime (Zeilen 140-164) und ftouch (Zeilen 166-179). Mit diesen Funktionen soll der Zeitpunkt der letzten Modifikation von Dateien bearbeitet werden. Der Zeitpunkt der letzten Modifikation ist die Zeit, die bei der Ausgabe von Verzeichnissen im Desktop angegeben wird, wenn die Verzeichnisausgabe in Form von "Text" geschieht. Auch hier verwendet GEMDOS das in der letzten Folge beschriebene Zeitformat. Allerdings existiert lediglich eine Funktion zum Setzen und Auslesen dieses Zeitwertes: Fdatime. Die Funktionen fgetdattime und fputdattime sind entsprechend beide auf Fdatime abzubilden. Dabei sind die gleichen Kodierungen bzw. Dekodierungen wie für das Setzen der Systemuhr vorzunehmen. Zusätzlich gilt es, die gewünschte Datei zunächst zu eröffnen, da Fdatime nicht direkt mit dem Dateinamen arbeitet, sondern stattdessen eine Kanalkennung benötigt. ftouch ist eine Variante von fputdattime. ftouch setzt das Datum der letzten Modifikation gerade auf die aktuelle Systemzeit. Damit ist es etwa möglich, nach Bearbeitung einer Datei das Modifikationsdatum zu aktualisieren. Zwei weitere Funktionen werden benötigt, um Dateien darauf zu überprüfen, ob es sich um "normale" Dateien (fexist, Zeilen 196-206) oder um Verzeichnisse (fisdir, Zeilen 208-219) handelt. Das Existenzkriterium für eine normale Datei besteht dabei darin, daß sich die Datei mit Fopen zum lesenden Zugriff öffnen läßt. Das Kriterium für ein Verzeichnis besteht darin, daß sich das aktuelle Verzeichnis mit acd (siehe unten) auf den gewünschten Pfad setzen läßt.

Die nächste Funktion im Listing, confirm (Zeilen 239-245), ist eine simple Hilfsfunktion. Sie führt einen kurzen Benutzerdialog der Form:

```
printf(text, name);
```

Erwartet wird entweder eine Bestätigung ('y') oder eine andere Eingabe. Das Er-

gebnis von confirm gibt dabei an, ob die Bestätigung positiv oder negativ ausgefallen ist. confirm wird von den drei nachfolgend beschriebenen Funktionen genutzt.

Eine etwas umfangreichere Funktion ist acp. Mit acp wird eine Datei mit dem Pfad src auf eine Datei mit dem Pfad dest kopiert. Existiert dest bereits, wird es dabei überschrieben. Um dies zu vermeiden, kann acp mit dem Wert TRUE für das Flag interactive aufgerufen werden. In diesem Fall führt acp eine Rückfrage durch, ob dest auch tatsächlich überschrieben werden soll. Weiterhin ist zu berücksichtigen, daß acp eine Datei nicht auf sich selbst kopieren darf (src = dest). In der Implementierung von acp (Listing 1.3, Zeilen 270-318) werden eine ganze Reihe von Betriebssystemaufrufen benutzt. Zunächst sind hier die beiden Aufrufe zur dynamischen Speicherverwaltung, Malloc und Mfree, zu nennen. Mit ihnen wird ein Schreib-/Lesepuffer zur Aufnahme der Kopierinformationen verwaltet. Die Verwendung des dynamischen Speichers erscheint dabei sinnvoll, da der benötigte Puffer nicht gerade klein ist. Die Konstante BSIZE gibt für ihn eine voreingestellte Größe von 32 kByte an. Der Kopiervorgang vollzieht sich in vier Phasen:

1. **Fehlerprüfung (Zeilen 282-287)**

   Die beiden Pfade werden auf Gleichheit überprüft, und es wird im gegebenen Fall (interactive-Flag ist gesetzt, und die Kopierung erfolgt auf eine existierende Datei) ein Benutzerdialog durchgeführt.

2. **Initialisierung (Zeilen 288-304)**

   Der Schreib-/Lesepuffer wird eingerichtet, und die Ein- und Ausgabekanäle werden geöffnet.

3. **Kopieren (Zeilen 305-311)**

   Der eigentliche Kern der Kopierroutine ist recht klein. Durch wiederholten, aufeinanderfolgenden Aufruf der beiden Betriebssystemaufrufe Fread und Fwrite wird die Quelldatei stückweise in die Zieldatei kopiert. Die Maximalgröße der Stücke ist dabei bereits durch die Puffergröße (BSIZE) vorgegeben.

4. **Terminierung (Zeilen 312-316)**

   Der Schreib-/Lesepuffer wird freigegeben, und die Datenkanäle werden geschlossen.

Auf diese etwas umfangreichere Funktion folgen nun wieder einige einfachere: Zunächst ist dies die Funktion arm (Zeilen 319-334). Sie realisiert das Löschen einer Datei. Unterschieden wird dabei zwischen dem Löschen von "normalen" und dem von schreibgeschützten Dateien. Bei den letztgenannten wird zusätzlich eine Sicherheitsabfrage durchgeführt, da hier im gegebenen Fall zunächst der Schreibschutz entfernt werden muß (Fattrib), weil sonst das Löschen (unlink) nicht möglich ist. Außerdem besteht wie bereits bei acp die Möglichkeit, Interaktion vorzuschreiben (interactive-Flag).

Die nächste Funktion - amv (Zeilen 336-344) - besitzt zwar eine ähnlich hohe Komplexität wie acp: amv realisiert das Bewegen einer Datei auf einen anderen Pfad. Allerdings ist die Implementierung weitaus einfacher, da amv sich leicht durch Hintereinanderausführung von acp und arm durchführen läßt.

Zwei weitere Funktionen - apwd (Zeilen 362-373) und acd (Zeilen 375-384) - sind wieder Abstraktionen von Funktionen auf Betriebssystemebene. Sie abstrahieren Dgetpath und Dsetpath, die das Lesen und Setzen des aktuellen Verzeichnisses ermöglichen. Die Funktionalität dieser beiden GEMDOS-Funktionen wird durch apwd und acd dahingehend erweitert, daß auch Pfade mit Gerätekennung beim Lesen und Setzen des Arbeitsverzeichnisses verwendet werden dürfen. Zur Anwendung kommen zwei weitere Betriebssystemaufrufe: Dgetdrv und Dsetdrv.

Der dritte Bereich des Listings 1.3, die Funktionen zur String-Bearbeitung, beginnt mit der Funktion pathsplit (Zeilen 401-446). pathsplit zerlegt den Pfad path in seinen Verzeichnis- (directory) und seinen Dateianteil (filename). Da dabei auch einige Grenzfälle berücksichtigt werden müssen, ergibt sich eine etwas umfangreichere, aber nicht besonders schwierige Funktion. Zur Anwendung gelangen einige der String-Funktionen aus der C-Bibliothek. Entsprechend macht die Funktion pathsplit den Header "STRING.H" erforderlich. Die Details sollen an dieser Stelle jedoch nicht diskutiert werden, da sie nicht weiter interessieren. Auch die weiteren Funktionen des Bereichs String-Bearbeitung sind alle nicht besonders schwierig, sie sollen nur kurz in ihrer Funktionalität dargestellt werden:

Die Funktion convupper (Zeilen 460-467) konvertiert alle Kleinbuchstaben des Strings str in Großbuchstaben.

Die Funktion ctop (Zeilen 486-502) konvertiert den nullterminierten C-String cstring in den Pascal-String pstring. Die Funktion ptoc (Zeilen 495-502) erledigt die umgekehrte Arbeit.

Anmerkung: Im Gegensatz zu den nur durch die Nullterminierung begrenzten C-Strings kann bei Pascal-Strings die Nullterminierung fehlen. Dafür besitzen sie anstelle des ersten Zeichens eine String-Längenangabe. Ein Nachteil dieses Datenformats besteht darin, daß Pascal-Strings maximal 256 Zeichen lang sein können. Dies ist gerade der maximale Wertebereich eines Zeichens. Pascal-Strings werden bei einigen Betriebssystemaufrufen benötigt, so z.B. bei Pexec.

Die letzte Funktion des Listings 1.3, onlyalpha (Zeilen 518-526), überprüft die Zeichenkette str darauf, ob sie nur aus Buchstaben besteht.

Wie eingangs erwähnt, sollen die Funktionen des Listings 1.3 im weiteren Ablauf der Programmierung als Hilfsfunktionen zur Verfügung stehen. Entsprechend ist die Datei "ATOM.C" als Modul zu compilieren, und es ist eine Header-Datei zu formulieren, die die Funktionsdeklarationen aus "ATOM.C" aufnimmt. Sinnvollerweise nennt man diesen Header dann auch "ATOM.H". "ATOM.H" finden Sie als Listing 1.4 abgedruckt.

Ein Makefile für den ersten Block

Apropos Compilierung von Modulen. Ich denke, nach der Implementierung des ersten Moduls ist es an der Zeit, das Makefile für den kompletten ersten Block der Toolbox abzudrucken. Es dürfte sich irgendwo hinter den heutigen Listings befinden. Das Makefile ist dabei auf die Belange des Laser C-Systems zugeschnitten. Die entsprechenden Kommandonamen und Parameter für Compiler und Linker sind jedoch als Makros definiert, so daß es kein Problem darstellen dürfte, das Makefile durch Umdefinition der Makros auf andere C-Systeme anzupassen. Voraussetzung für eine Anpassung ist natürlich ein C-System mit MAKE-Utility.

Vorausschau

Das war's dann auch mal wieder für heute. In der nächsten Folge der "Programmer's Toolbox" gelangen wir (endlich) an den Punkt, wo erste Kommandos programmiert werden. Es handelt sich dabei um:

CD	-	Ändern des Arbeitsverzeichnisses
PWD	-	Ausgabe des Arbeitsverzeichnisses
DATE	-	Auslesen/Setzen des Systemdatums
TIME	-	Auslesen/Setzen der Systemzeit
DF	-	Auslastungsstatistik für Dateisysteme

Dirk Brockhaus

```
  1:  /*
  2:   * Listing 1.3, Datei : atom.c
  3:   * Modul              : ATOM - ATOMare Manipulat.
  4:   * Modifikationsdatum : 08-Nov-89
  5:   * Abhängigkeiten     : stdio.h,ctype.h,string.h,
  6:   *                      osbind.h, local.h
  7:   */
  8:
  9:  #include <stdio.h>
 10:  #include <ctype.h>
 11:  #include <string.h>
 12:  #include <osbind.h>
 13:  #include "local.h"
 14:
 15:  /*
 16:   * Funktionen    : getdate, putdate, gettime,
                       puttime
 17:   *
 18:   * Parameter     : getdate(&day, &month, &year);
 19:   *                 ok = putdate(day, month, year);
 20:   *                      gettime(&hour, &minute,
                                   &second);
 21:   *                 ok = puttime(hour, minute,
                                   second);
 22:   *                 short   day,
 23:   *                         month,
 24:   *                         year,
 25:   *                         hour,
 26:   *                         minute,
 27:   *                         second;
 28:   *                 BOOLEAN ok;
 29:   *
 30:   * Aufgabe       :
 31:   *
 32:   * Das Systemdatum (date) bzw. die Systemzeit
         (time)
 33:   * wird in der Auflösung <day>, <month>, <year>,
 34:   * <hour>, <minute>, <second> gelesen (get) bzw.
 35:   * geschrieben (put). <ok> gibt dabei an, ob ein
 36:   * Schreibvorgang erfolgreich durchgeführt werden
 37:   * konnte.
 38:   */
 39:
 40:  void getdate(day, month, year)
 41:  short *day,
 42:        *month,
 43:        *year;
 44:  {   unsigned date = Tgetdate();
 45:
 46:      *year = ((date & 0xfe00) >> 9) + 1980;
 47:      *month = (date & 0x01e0) >> 5;
 48:      *day = (date & 0x001f);
 49:  }
 50:
 51:  BOOLEAN putdate(day, month, year)
 52:  short day,
 53:        month,
 54:        year;
 55:  {   unsigned date;
 56:
 57:      date = ((unsigned)(year - 1980) << 9) |
 58:             ((unsigned)(month) << 5) |
 59:              (unsigned)(day);
 60:      return(Tsetdate(date) == 0);
 61:  }
 62:
 63:  void gettime(hour, minute, second)
 64:  short *hour,
 65:        *minute,
 66:        *second;
 67:  {   unsigned time = Tgettime();
 68:
 69:      *hour = (time & 0xf800) >> 11;
 70:      *minute = (time & 0x07e0) >> 5;
 71:      *second = (time & 0x001f) * 2;
 72:  }
 73:
 74:  BOOLEAN puttime(hour, minute, second)
 75:  short hour,
 76:        minute,
 77:        second;
 78:  {   unsigned time;
 79:
 80:      time = ((unsigned)(hour) << 11) |
 81:             ((unsigned)(minute) << 5) |
 82:              (unsigned)(second / 2);
 83:      return(Tsettime(time) == 0);
 84:  }
 85:
 86:  /*
 87:   * Funktionen    : fgetdattime, fputdattime,
       ftouch
 88:   *
 89:   * Parameter     : ok = fgetdattime(filename,
                                      &day,
 90:   *                                &month, &year,
                                      &hour,
 91:   *                                &minute,
                                      &second);
 92:   *                 ok = fputdattime(filename, day,
 93:   *                                month, year,
                                      hour,
 94:   *                                minute,
                                      second);
 95:   *                 ok = ftouch(filename);
 96:   *                 char  *filename;
 97:   *                 short day,
 98:   *                       month,
 99:   *                       year,
100:   *                       hour,
101:   *                       minute,
102:   *                       second;
103:   *
104:   * Aufgabe       :
105:   *
106:   * Das Modifikationsdatum einer Datei mit dem
         Pfadnamen
107:   * <filename> wird in der Auflösung <day>,
         <month>,
108:   * <year>, <hour>, <minute>, <second> gelesen
         (get),
109:   * geschrieben (put) oder aktualisiert (touch).
         <ok>
110:   * gibt an, ob die gewünschte Operation
         erfolgreich
111:   * durchgeführt werden konnte.
112:   */
113:
114:  BOOLEAN fgetdattime(filename, day, month, year,
115:                      hour, minute, second)
116:  char  *filename;
117:  short *day,
118:        *month,
119:        *year,
120:        *hour,
121:        *minute,
122:        *second;
123:  {   unsigned time[2];
124:      int      handle;
125:
126:      handle = Fopen(filename, 0);
127:      if (handle <= 0)
128:          return(FALSE);
129:      Fdatime(time, handle, 0);
130:      Fclose(handle);
131:      *hour = (time[0] & 0xf800) >> 11;
132:      *minute = (time[0] & 0x07e0) >> 5;
133:      *second = (time[0] & 0x001f) * 2;
134:      *year = ((time[1] & 0xfe00) >> 9) + 1980;
135:      *month = (time[1] & 0x01e0) >> 5;
136:      *day = (time[1] & 0x001f);
137:      return(TRUE);
138:  }
139:
140:  BOOLEAN fputdattime(filename, day, month, year,
141:                      hour, minute, second)
142:  char  *filename;
143:  short day,
144:        month,
145:        year,
146:        hour,
147:        minute,
148:        second;
149:  {   unsigned time[2];
150:      int      handle;
151:
152:      handle = Fopen(filename, 0);
153:      if (handle <= 0)
154:          return(FALSE);
155:      time[0] = ((unsigned)(hour) << 11) |
156:                ((unsigned)(minute) << 5) |
157:                (unsigned)(second / 2);
158:      time[1] = ((unsigned)(year - 1980) << 9) |
159:                ((unsigned)(month) << 5) |        →
```

DISKETTEN-SERVICE

Absender bitte deutlich schreiben

Vorname / Name

Straße / Nr.

PLZ / Ort

Telefon

Postkarte

Bitte freimachen

MAXON Computer
Schwalbacher Straße 52

D-6236 Eschborn

ST COMPUTER

Kleinanzeigen

ST COMPUTER

Kleinanzeigen

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte freimachen

Heim Verlag

Heidelberger Landstr. 194

6100 Darmstadt 13



ABO



Bitte freimachen

Postkarte



Heidelberger Landstr. 194

6100 Darmstadt 13



Bitte senden Sie mir folgende PD-Disketten / Sonderdisks

Datum / Unterschrift

Zahlung erfolgt:

___ Vorauskasse (per Scheck) ___ per Nachnahme

- Pro PD-Diskette entfällt ein Unkostenbeitrag von DM 10.-
- Preis der Sonderdisks je nach Angabe
- Bei Nachnahme zuzüglich DM 4.- Nachnahmegebühr
- Versandkosten (Porto und Verpackung) DM 5.- (Ausland DM 10.-)
- Auslandsbestellungen nur gegen Vorauskasse möglich

Ab 5 Disketten versandkostenfrei

SONDERDISK

Bitte beachten Sie, daß Sonderdisks ein Copyright besitzen und nicht weiterkopiert werden dürfen.

DISKETTEN-SERVICE



Kleinanzeigen-Auftrag

Bitte veröffentlichen Sie für mich folgende Kleinanzeige in der angekreuzten Rubrik

Biete an Hardware / Software **Ich suche** Hardware / Software **Tausch** **Kontakte** Verschiedenes

30 Buchstaben je Standardzeile – incl. Satzzeichen und Wortzwischenräume
Groß- und Kleinbuchstaben verwenden, fettgedruckte Wörter unterstreichen

Bearbeitung nur **gegen Vorausscheck** über den entsprechenden Betrag (keine Überweisung)

privat = DM 4.- je Zeile
gewerblich = DM 12.- je Zeile
Chiffregebühr = DM 10.-

Scheck über DM ______
ist beigefügt

Bei Angeboten: Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze

Datum ______ Unterschrift ______

Absenderangaben auf der Rückseite nicht vergessen



Kleinanzeigen

ST COMPUTER

Abonnement

Kd.-Nr.:

Ja, bitte senden Sie mir die ATARI-Computer Fachzeitschrift ab der nächsten erreichbaren Ausgabe für mindestens 1 Jahr (11 Hefte) zum ermäßigten Preis von jährlich DM 80,- frei Haus.
(Ausland: nur gegen Scheck-Voreinsendung DM 100,- Normalpost, DM 130,- Luftpost)
Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Geschenk- bzw. Lieferadresse:

Vorname

Name

Straße/Postfach

Land PLZ Ort

Gewünschte Zahlungsweise bitte ankreuzen

Bequem und bargeldlos durch Bankeinzug

Konto-Nr. BLZ

Institut Ort

☐ Ein Verrechnungsscheck über DM ______ liegt bei.

☐ Vorauskasse per Zahlung auf unser Postscheck-Konto Ffm. BLZ 500 100 60, Kto.-Nr. 5537-602

Diese Vereinbarung kann ich innerhalb von 8 Tagen beim Heim-Verlag, Heidelberger Landstr. 194, 6100 Darmstadt-Eberstadt widerrufen. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs. Ich bestätige die Kenntnisnahme des Widerrufsrechts durch meine 2. Unterschrift.

Datum Unterschrift

Datum Unterschrift

ST COMPUTER

ABO

Diese Vereinbarung kann ich innerhalb von 8 Tagen beim Heim-Verlag, Heidelberger Landstr. 194, 6100 Darmstadt-Eberstadt widerrufen. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs. Ich bestätige die Kenntnisnahme des Widerrufsrechts durch meine 2. Unterschrift.

Buch-Neuheiten



Pascal auf dem Atari ST

Band 1: Das Einsteiger-Buch
Band 2: Das Profi-Buch

Haben Sie genug vom Spaghetti-Code unübersichtlicher Basic-Programme? Hat Ihr Monitor Kopfschmerzen von den Bomben fehleranfälliger C-Programme? Dann sollten Sie PASCAL kennenlernen!

PASCAL ermöglicht:
- klar gegliederte und strukturierte Programme
- flexible und selbstdefinierte Datentypen.

ST PASCAL PLUS ermöglicht darüber hinaus:
- einfachen Zugriff auf sämtliche Betriebssystemroutinen
- komfortable Nutzung der graphischen Benutzeroberfläche GEM.

In zwei Bänden machen wir Sie vom PASCAL-Einsteiger zum Profi.

Band 1
Hardcover, über 400 Seiten
Inkl. Diskette
Best Nr. B-439
ISBN 3-923250-81-9

DM 59,-

„Das Einsteiger-Buch" (Band 1) fängt bei Null an und setzt keinerlei Vorkenntnisse voraus. Es stellt eine umfassende und leichtverständliche Einführung dar. Übungsaufgaben am Ende jedes Kapitels bieten dem Leser die Möglichkeit, die gewonnenen Erkenntnisse in die Praxis umzusetzen. Vergleiche zu Basic erleichtern ehemaligen Basic-Programmierern den Umstieg. Besondere Beachtung erfährt das von CCD vertriebene ST PASCAL PLUS-SYSTEM. Sollten Sie schon einen anderen Compiler benutzen, können Sie ohne weiteres mit diesem Buch arbeiten, da alle Abweichungen vom PASCAL-ISO-STANDARD vermerkt sind.

Grundlagen: • Algorithmen, Programme – Maschinensprache, Assembler und höhere Programmiersprachen – Dolmetscherprogramme: Compiler und Interpreter, Linker • Die Arbeit mit dem ST PASCAL PLUS-System • Installation und Handhabung einer RAM-Disk • Variablen, Datentypen, Konstanten • Kontrollstrukturen: IF...THEN...ELSE, CASE...OF • Boole'sche Algebra: AND, OR, NOT • Schleifen: REPEAT...UNTIL, WHILE...DO, FOR...TO...DO, LOOP...END • Prozeduren und Funktionen – Geltungsbereiche: globale und lokale Variablen – formale und variable Parameter • ST Intern: RAM und ROM, Zahlensysteme, Bits und Bytes – Interne Darstellung von PASCAL-Datentypen - Bitoperationen • Deklaration eigener Typen • Mengen (Sets) • Mengenoperationen: Schnittmenge, Vereinigungsmenge, Mengendifferenz • Felder (Arrays) • Zeichenketten (Strings) – Unterschiede von PASCAL- und C-Strings – Umwandlung String < – > Zahl • Verbunde (Records) • Dateien (Files): GEMDOS-Pfadnamen – Dateioperationen: RESET, REWRITE, CLOSE, RENAME ERASE – Textdateien.

Band 2
Hardcover
über 400 Seiten
Bestell-Nr. 444

DM 59,-

Alle genannten Preise
sind unverbindliche Verkaufspreise



Professionelles Programmieren in C

schließt die bisher existierende Lücke zwischen Einsteiger- und komplexer Fachliteratur bzw. Nachschlagewerken. Es listet keine Funktionen auf mit lieblosen Beispielen, sondern zeigt anhand zweier professioneller Programme die effektive Entwicklung von Software. Der Autor, mehrfacher Gewinner von Programmierwettbewerben und freier Mitarbeiter mehrerer Softwarehäuser, vermittelt auch Hintergrundwissen über die Entstehung eines Programms und gibt Tips, wie Sie selbst erfolgreich Programme schreiben können. Der hochkomplexe Stoff wird in leicht verständlicher Sprache und mit gut dokumentierten Beispielen behandelt. Das Buch läßt Sie auch große Beispielprogramme verstehen und ähnliche Programme selbst erstellen.

Hardcover, über 400 Seiten
Inkl. Diskette
Best Nr. B-438
ISBN 3-923250-78-9

DM 59,-

Aus dem Inhalt: • Konzepte zur Datenorganisation • Dynamische Speicherverwaltung • Verkettung über Pointer • Rekursionen • Universelle und schnelle Sortierverfahren • Universelle Druckeranpassung und Seitenformatierung • Der leichte GEM-Einstieg: Drop-Down-Menüs • In den Tiefen von GEM: Objekte und Formulare • Fenstertechnik mal anders betrachtet. Es wird nicht nur „Wissen pur" vermittelt: • Der Weg zum Programm (das Konzept, Top-Down-Analyse, die richtigen Tools) • Professionelle Programmierung (Dokumentation, Modularisierung, Datensicherheit, Benutzerführung) • Wenig Aufwand, große Wirkung (Abschätzung von Aufwand und Nutzen, Nutzung fremder Routinen).

C auf dem ATARI ST

„C" ist die zweite „Muttersprache" des ATARI ST: schnell, komfortabel, kompakt im Code. Das Buch „C auf dem ATARI ST" ist für Anwender geeignet, die bereits Erfahrungen mit anderen Programmiersprachen gemacht haben. Es behandelt den vollen Sprachumfang von Standard-C und verweist auf BASIC und PASCAL. Besonderer Wert wird auf die Anschaulichkeit und Genauigkeit der Darstellung gelegt. Alle Programmbeispiele sind getestet und direkt in den Text übernommen. In C lassen sich einfache Programme von außerordentlicher Übersichtlichkeit und Klarheit schreiben. Dieses Buch vermittelt eine solide Grundlage für die Programmierung in dieser eleganten Sprache. Die besonderen Fähigkeiten des ATARI ST kommen dabei nicht zu kurz. Alle Programmbeispiele enthält die Programmdiskette.



Inkl. Diskette
Best Nr. B-406

DM 59,-

Aus dem Inhalt: • Die Programmiersprache • Programmierung mit Editor und Compiler • C-Compiler für ATARI: Digital Research, Lattice, Megamax • Grundlegende Elemente eines C-Programms • Variablentypen • Felder und Vektoren • Diskettenhandling • Einbindung von Assemblerprogrammen • Fehler in C-Compilern • Tools und anderes.

BESTELL-COUPON

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Bitte senden Sie mir: ____

Pascal I	à 59,- DM
Pascal II	à 59,- DM
C auf dem ATARI ST	à 59,- DM
Professionelles Programmieren in C	à 59,- DM

Name, Vorname ____
Straße, Hausnr. ____
PLZ, Ort ____

Oder benutzen Sie die eingeheftete Bestellkarte (zzgl. Versandkosten DM 6,- Ausland DM 10,-)

In der Schweiz:
Data Trade AG
Landstr. 1
CH - 5415 Rieden-Baden

```
160:                (unsigned)(day);
161:        Fdatime(time, handle, 1);
162:        Fclose(handle);
163:        return(TRUE);
164:    }
165:
166:    BOOLEAN ftouch(filename)
167:    char *filename;
168:    {   short day,
169:              month,
170:              year,
171:              hour,
172:              minute,
173:              second;
174:
175:        getdate(&day, &month, &year);
176:        gettime(&hour, &minute, &second);
177:        return(fputdattime(filename, day, month,
            year,
178:                        hour, minute, second));
179:    }
180:
181:    /*
182:     * Funktion     : fexist, fisdir
183:     *
184:     * Parameter    : ok = fexist(filename);
185:     *                ok = fisdir(filename);
186:     *                BOOLEAN ok;
187:     *                char    *filename;
188:     *
189:     * Aufgabe      :
190:     *
191:     * Es wird überprüft, ob die Datei mit dem Namen
192:     * <filename> existiert (exist) bzw. ein
       Verzeichnis
193:     * ist (isdir).
194:     */
195:
196:    BOOLEAN fexist(filename)
197:    char *filename;
198:    {   int     handle;
199:        BOOLEAN ok;
200:
201:        handle = Fopen(filename, 0);
202:        ok = handle > 0;
203:        if (ok)
204:            Fclose(handle);
205:        return(ok);
206:    }
207:
208:    BOOLEAN fisdir(filename)
209:    char *filename;
210:    {   void    apwd(),
211:        BOOLEAN acd(),
212:                ok;
213:        char    actdir[100];
214:
215:        apwd(actdir);
216:        ok = acd(filename);
217:        acd(actdir);
218:        return(ok);
219:    }
220:
221:    /*
222:     * Funktion     : confirm
223:     *
224:     * Parameter    : ok = confirm(text, name);
225:     *                BOOLEAN ok;
226:     *                char    *text,
227:     *                        *name;
228:     *
229:     * Aufgabe      :
230:     *
231:     * Hilfsfunktion. Es wird ein Benutzerdialog
       über die
232:     * Standardkanäle geführt, wobei <text> als
       Information
233:     * mit dem Stringparameter <name> ausgegeben
       wird.
234:     * <confirm> liefert abhängig von seiner Eingabe
       einen
235:     * Wahrheitswert (TRUE für die Eingabe "y", FALSE
236:     * sonst).
237:     */
238:
239:    BOOLEAN confirm(text, name)
240:    char *text,
241:         *name;
242:    {   printf(text, name);
243:        fflush(stdout);
244:        return(Cconin() == 'y');
245:    }
246:
247:    /*
248:     * Funktionen   : acp, arm, amv
249:     *
250:     * Parameter    : ok = acp(src, dest,
                            interactive);
251:     *                ok = arm(name, interactive);
252:     *                ok = amv(oldname, newname);
253:     *                char    *src,
254:     *                        *dest,
255:     *                        *name,
256:     *                        *oldname,
257:     *                        *newname;
258:     *                BOOLEAN interactive;
259:     *
260:     * Aufgabe      :
261:     *
262:     * Das Kopieren (acp), Löschen (arm) oder
       Bewegen (amv)
263:     * von einzelnen Dateien. <ok> gibt dabei, an ob
       die
264:     * gewünschte Operation mit Erfolg durchgeführt
       werden
265:     * konnte. <interactive> gibt an, ob bei
       Datenverlust
266:     * durch die gewünschte Operation beim Anwender
267:     * zurückgefragt werden soll.
268:     */
269:
270:    #define BSIZE 32768L  /* PuffergröPe beim
                              Kopieren */
271:
272:    BOOLEAN acp(src, dest, interactive)
273:    char    *src,
274:            *dest;
275:    BOOLEAN interactive;
276:    {   int     input,
277:                output;
278:        long    count,
279:                buf;
280:        BOOLEAN ok;
281:
282:        if (strcmp(src, dest) == 0)
283:            return(FALSE);
284:        if (interactive && fexist(dest))
285:            if (!confirm("\n%s already exists.
                        Proceed? ",
286:                dest))
287:                return(TRUE);
288:        buf = Malloc(BSIZE);
289:        input = Fopen(src, 0);
290:        if (input <= 0) {
291:            Mfree(buf);
292:            return(FALSE);
293:        }
294:        output = Fcreate(dest, 0);
295:        if (output <= 0) {
296:            Mfree(buf);
297:            return(FALSE);
298:        }
299:        Fclose(output);
300:        output = Fopen(dest, 1);
301:        if (output <= 0) {
302:            Mfree(buf);
303:            return(FALSE);
304:        }
305:        do {
306:            count = Fread(input, BSIZE, buf);
307:            if(Fwrite(output,count,buf)!=count) {
308:                Mfree(buf);
309:                return(FALSE);
310:            }
311:        } while (count > 0);
312:        Mfree(buf);
313:        ok=Fclose(output) == 0 && Fclose(input) == 0;
314:        if (ok)
315:            Fattrib(dest, 1, Fattrib(src, 0, 0));
316:        return(ok);
317:    }
318:
319:    BOOLEAN arm(name, interactiv)
320:    char    *name;
```

→

```
321:  BOOLEAN interactiv;
322:  {   if (interactiv && !confirm("\n%s? ", name))
323:          return(TRUE);
324:      if (fexist(name)) {
325:          if (Fattrib(name, 0, 0) & 0x01)
326:            if (confirm(
327:              "\nOverwrite protection for %s.
               Proceed? ",
328:              name))
329:                Fattrib(name, 1, 0x00);
330:          return(unlink(name) == 0);
331:      }
332:      else
333:          return(FALSE);
334:  }
335:
336:  BOOLEAN amv(oldname, newname)
337:  char *oldname,
338:       *newname;
339:  {   if (strcmp(oldname, newname) == 0)
340:          return(FALSE);
341:      if (!acp(oldname, newname, TRUE))
342:          return(FALSE);
343:      return(arm(oldname, TRUE));
344:  }
345:
346:  /*
347:   * Funktionen    : apwd, acd
348:   *
349:   * Parameter     : apwd(directory);
350:   *                 ok = acd(directory);
351:   *                 char    *directory;
352:   *                 BOOLEAN ok;
353:   *
354:   * Aufgabe       :
355:   *
356:   * Das Lesen (apwd) und Setzen (acd) des
       aktuellen
357:   * Verzeichnisses <directory>. <ok> gibt an ob
       die
358:   * gewünschte Operation erfolgreich durchgeführt
       werden
359:   * konnte.
360:   */
361:
362:  void apwd(directory)
363:  char *directory;
364:  {   char drv[2],
365:           path[100];
366:
367:      drv[0] = Dgetdrv() + 'A';
368:      drv[1] = 0;
369:      Dgetpath(path, 0);
370:      if (strlen(path) == 0)
371:          strcat(path, "\\");
372:      sprintf(directory, "%s:%s", drv, path);
373:  }
374:
375:  BOOLEAN acd(directory)
376:  char *directory;
377:  {   if (isupper(directory[0]) &&
378:          directory[1] == ':')
379:          Dsetdrv((short)(directory[0] - 'A'));
380:      else if (islower(directory[0]) &&
381:               directory[1] == ':')
382:          Dsetdrv((short)(directory[0] - 'a'));
383:      return(Dsetpath(directory) == 0);
384:  }
385:
386:  /*
387:   * Funktion      : pathsplit
388:   *
389:   * Parameter     : pathsplit(path, directory,
                                 filename);
390:   *                 char *path,
391:   *                      *directory,
392:   *                      *filename;
393:   *
394:   * Aufgabe       :
395:   *
396:   * Ein Pfadname <path> wird aufgespalten in
       seinen
397:   * Pfadanteil <directory> und seinen Dateianteil
398:   * <filename>.
399:   */
400:
401:  void pathsplit(path, directory, filename)
402:  char *path,
403:       *directory,
404:       *filename;
405:  {   int     pathlen = strlen(path),
406:              startfile;
407:      char    pathcopy[100],
408:              *help;
409:      BOOLEAN start = TRUE;
410:
411:      strcpy(pathcopy, path);
412:      startfile = pathlen - 1;
413:      while (startfile >= 0 && (isalpha(pathcopy[
414:             startfile]) ||
415:             isdigit(pathcopy[startfile]) ||
416:             (pathcopy[startfile] == '.' &&
                                      !start) ||
417:             pathcopy[startfile] == '*' ||
418:             pathcopy[startfile] == '?' ||
419:             pathcopy[startfile] == '_')) {
420:          startfile--;
421:          start = FALSE;
422:      }
423:      if (startfile < 0) {
424:          strcpy(directory, "");
425:          strcpy(filename, pathcopy);
426:      }
427:      else if (startfile == pathlen - 1) {
428:          strcpy(directory, pathcopy);
429:          strcpy(filename, "");
430:          if(directory[strlen(directory)-1]!='\\')
431:              strcat(directory, "\\");
432:      }
433:      else {
434:          help = &pathcopy[startfile + 1];
435:          strcpy(filename, help);
436:          do {
437:              pathcopy[startfile + 1] = 0;
438:              startfile--;
439:          } while (startfile >= 0 &&
440:                  !(isalpha(pathcopy[startfile +
                          1]) ||
441:                  isdigit(pathcopy[startfile]) ||
442:                  pathcopy[startfile+1]==':' ||
443:                  pathcopy[startfile+1]=='.'));
444:          strcpy(directory, pathcopy);
445:          strcat(directory, "\\");
446:      }
447:  }
448:
449:  /*
450:   * Funktion      : convupper
451:   *
452:   * Parameter     : convupper(str);
453:   *                 char *str;
454:   *
455:   * Aufgabe       :
456:   *
457:   * convupper konvertiert die Kleinbuchstaben
458:   * innerhalb des Strings <str> in Großbuchstaben.
459:   */
460:
461:  void convupper(str)
462:  char *str;
463:  {   short i;
464:
465:      for (i = 0; i < strlen(str); i++)
466:          if (islower(str[i]))
467:              str[i] = _toupper(str[i]);
468:  }
469:
470:  /*
471:   * Funktion      : ctop, ptoc
472:   *
473:   * Parameter     : ctop(cstr, pstr);
474:   *                 ptoc(pstr, cstr);
475:   *                 char *cstr,
476:   *                      *pstr;
477:   *
478:   * Aufgabe       :
479:   *
480:   * Konvertierung von C-Strings in Pascal-Strings
481:   * und umgekehrt. <ctop> und <ptoc> werden an der
482:   * Schnittstelle zu einigen
       Betriebssystemaufrufen
483:   * benötigt (etwa bei <Pexec>). Hier werden
       teilweise
484:   * Pascal-Strings von seiten des ST verwendet. →
```

```
485:    */
486:
487:  void ctop(cstring, pstring)
488:  char *cstring, *pstring;
489:  {    short i;
490:
491:       pstring[0] = strlen(cstring);
492:       for (i = 1; i <= pstring[0]; i++)
493:           pstring[i] = cstring[i - 1];
494:  }
495:
496:  void ptoc(pstring, cstring)
497:  char *pstring, *cstring;
498:  {    short i;
499:
500:       for (i = 0; i < pstring[0]; i++)
501:           cstring[i] = pstring[i + 1];
502:       cstring[pstring[0]] = 0;
503:  }
504:
505:  /*
506:   * Funktion      : onlyalpha
507:   *
508:   * Parameter     : isonly = onlyalpha(str);
509:   *                 BOOLEAN isonly;
510:   *                 char    *str;
511:   *
512:   * Aufgabe       :
513:   *
514:   * Der String <str> wird darauf überprüft, ob er
515:   * nur Buchstaben enthält. Ein entsprechender
516:   * Wahrheitswert wird als Ergebnis berechnet.
517:   */
518:
519:  BOOLEAN onlyalpha(str)
520:  char *str;
521:  {    short i;
522:
523:       for (i = 0; i < strlen(str); i++)
524:           if (!isalpha(str[i]) || str[i] == '_')
525:               return(FALSE);
526:       return(TRUE);
527:  }
```

```
 1:  /*
 2:   * Listing 1.4, Datei : atom.h
 3:   * Modul              : ATOM - ATOMare
                             Manipulationen
 4:   * Modifikationsdatum : 08-Nov-89
 5:   * Abhängigkeiten     : local.h
 6:   * (c) MAXON Computer GmbH 1990 /
 7:
 8:  extern void    getdate();
 9:  extern BOOLEAN putdate();
10:  extern void    gettime();
11:  extern BOOLEAN puttime(),
12:                 fgetdattime(),
13:                 fputdattime(),
14:                 ftouch(),
15:                 fexist(),
16:                 fisdir(),
17:                 acp(),
18:                 arm(),
19:                 amv();
20:  extern void    apwd();
21:  extern BOOLEAN acd();
22:  extern void    pathsplit(),
23:                 convupper(),
24:                 ctop(),
25:                 ptoc();
```



Ihr Buch-Konzept
Ihre Programm-Idee
Ihr Hardware-Projekt

Für unser großes Angebot an Büchern, Soft- und Hardware

HEIM-VERLAG-Bücher

Bücher wie **Programmieren in Omikron Basic, CAD - Praxis mit Campus** oder **C auf dem Atari ST** erleichtern den Anwendern und den Programmierern das Arbeiten mit Ihrem ST. Haben auch Sie eine Buchidee, dann sind wir der richtige Verlag für Sie !

HEIM-VERLAG-Software

Programme wie **That's Write, Steuer-Tax** oder **Skyplot** sind Ihnen garantiert ein Begriff. Sie bereiten den ST-Besitzern viel Freude. Arbeiten auch Sie an einem Software-Produkt ? Haben Sie Ihr Programm bereits in einer (Vor-)Version fertig? Dann sollten Sie sich unbedingt mit uns in Verbindung setzen !!

HEIM-VERLAG-Hardware

Wer einen ATARI ST besitzt und sich für Hardware interessiert, kommt nicht um den HEIM-VERLAG herum. Gerade in letzter Zeit haben wir mit unseren Hardware-Projekten für Aufregung unter den ST-Besitzern gesorgt. Produkte wie **PC- Speed, AT-Speed** oder **Scanface** sprechen für sich. Haben Sie also ein Hardware-Projekt in Planung, sind wir der richtige Partner für Sie !!!

Wir sichern...

... Ihnen eine leistungsfähige Vermarktung Ihres Programms oder Projekts mit einer attraktiven Umsatzbeteiligung zu. Sie nehmen somit direkt am Erfolg Ihres Produktes teil !!!!

Senden Sie uns...

... Ihr Konzept, Vor- oder Endversion Ihres Buches, Programms oder Hardware-Projekts zu und erläutern Sie kurz dessen Fähigkeiten. Wir setzen uns dann umgehend mit Ihnen in Verbindung. Wenn Sie jetzt noch Fragen haben, rufen Sie doch einfach mal bei uns an.

Heim Verlag

Heidelberger-Landstr. 194

6100 Darmstadt 13

Tel.: 06151 / 56057 oder

06151 / 595946

SciGraph 2.0

SciGraph 2.0 Highlights:

- [illegible]
- [illegible]
- Raster- und Farbverläufe
- Bézierkurven
- erweiterte Zeichenfunktionen
- neuer Tabelleneditor
- geringerer Speicherbedarf
- Postscript-, CVG- u. GEM-Ausgabe

ST-Statistik

Die ideale Ergänzung zu SciGraph 2.0

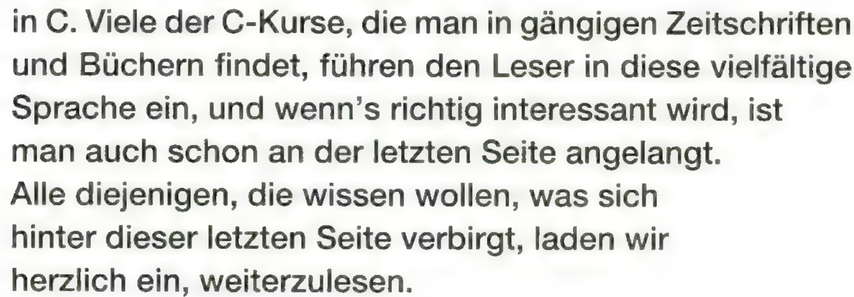
- Uni- und multivariante Verfahren von Crosstabs bis zur Clusteranalyse
- Grafiken in 2D und 3D mit Grafikeditor, Schnittstellen zu Text- und Grafikprogrammen, sowie Datenbanken

SciLab

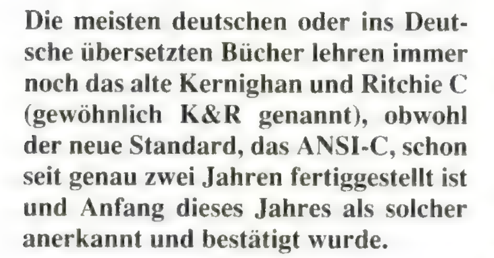
SciGraph 2.0 kostet	DM 599,–
ST-Statistik kostet	DM 348,–
Demo	DM 30,–

SciLab GmbH · Isestraße 57 · 2000 Hamburg 13
Hotline: 040 - 4 60 37 02 (tägl. 11⁰⁰ bis 13⁰⁰) Fax: 040 - 47 93 44

© M.V. ZIMMERMANN

Unsere neue Artikelreihe befaßt sich mit der Weiterführung der Programmierung

in C. Viele der C-Kurse, die man in gängigen Zeitschriften und Büchern findet, führen den Leser in diese vielfältige Sprache ein, und wenn's richtig interessant wird, ist man auch schon an der letzten Seite angelangt. Alle diejenigen, die wissen wollen, was sich hinter dieser letzten Seite verbirgt, laden wir herzlich ein, weiterzulesen.

1 ANSI-C


Die meisten deutschen oder ins Deutsche übersetzten Bücher lehren immer noch das alte Kernighan und Ritchie C (gewöhnlich K&R genannt), obwohl der neue Standard, das ANSI-C, schon seit genau zwei Jahren fertiggestellt ist und Anfang dieses Jahres als solcher anerkannt und bestätigt wurde.

Die Entwicklung ist nicht stehengeblieben. Die objektorientierte Programmierung (OOP) ist im Begriff, die Welt zu erobern, aber am ST programmieren viele noch im K&R-C. Der Grund dafür liegt sicher auch bei den Herstellern, die ihre sicher guten Compiler nicht oder zu langsam weiterentwickeln. Noch trauriger wird die Geschichte, wenn dieselben Compiler auf anderen 68000 Computern längst ANSI-C beherschen und man weiß, daß sich die ST-Portierung im Gegensatz z.B. zum AMIGA sehr einfach realisieren läßt. Dem ST-Programmierer bleiben, entscheidet er sich für ANSI-C, nur drei Möglichkeiten: Prospero C, GNU C oder Turbo C. Der zweite Kandidat kommt aus der Public Domain-Ecke und stellt am ST eher eine Compiler-Studie dar, die mit vielen Optimierungen arbeitet, dafür aber zuviel Speicher verbraucht (mehr dazu in einer der nächsten Folgen). Da sich Turbo C unter Entwicklern sehr bewährt hat, wird Prospero sich bemühen müssen, seinen Compiler auf dem ST-Markt zu etablieren. In meiner Serie verwende ich Turbo C V1.0 bis V2.0; jedoch sollte es für Sie leicht sein, mit einem anderen ANSI-fähigen Compiler zu arbeiten. Werfen wir nun einen Blick auf die wichtigsten ANSI-Eigenschaften.

Der Präprozessor

Der Präprozessor ist von Grund auf standardisiert und um wichtige Eigenschaften erweitert worden. Die im täglichen Gebrauch oft angewandte Befehlsfolge:

```
#else
#if
```

wurde von '#elif', das weitaus eleganter und vor allem übersichtlicher ist, ersetzt. Die von Pascal her bekannten Schalter, die verschiedene Optionen des Compilers steuern, wurden wie in ADA '#pragma' genannt. Sollten diese von verschiedenen Übersetzern nicht verstanden werden, werden sie einfach ignoriert. In Turbo C haben Sie damit die Möglichkeit, Warnungen ein- bzw. auszuschalten. Wenn Sie ein K&R-C-Programm übersetzen wollen, ohne gleich mit Warnungen über fehlende Prototypen überhäuft zu werden, dann setzen Sie

```
#pragma -W-pro
```

an den Anfang Ihres Programms. '#error' weist den Compiler an, die dem Befehl folgende Fehlermeldung auszugeben und den Übersetzerlauf zu stoppen. Sinnvoll ist diese Direktive nur mit bedingter Compilierung.

```
#if __TURBOC__ < 0x200
#     error "Läuft nur mit TC 2.0"
#endif
```

Die Leerstellen vor und hinter einem '#' sind ausdrücklich erlaubt, da sie der besseren Formatierung des Textes dienen. '#' hat jedoch auch eine neue Bedeutung bekommen: Es wird bei dem 'stringizing' angewandt, um bei der Expandierung eines Makros dessen Parameter als String auszugeben. Eine mögliche Anwendung sieht so aus:

```
#define VORNAME Grischa
#define text(string) #string
printf(text(VORNAME));
```

Nach der Ersetzung steht in der letzten Zeile

```
printf("Grischa");
```

Mit Hilfe von '##' ist es möglich, zwei Parameter zu einem Token verschmelzen zu lassen.

```
#define macro(a, b) a ## b
macro(re, turn);
```

ergibt wirklich den Befehl

```
return;
```

Achten Sie darauf, daß Makros innerhalb einer String-Konstanten nur ersetzt werden, wenn ihnen ein '#' vorangestellt wird. D.h.

```
#define RANK expert
printf("rank: #RANK");
```

Das 'defined()' wurde als ein gültiger Operator in den Standard aufgenommen. Kombiniert mit '#if' entspricht es dem '#ifdef', erlaubt aber Mehrfachüberprüfungen, da auch logische Operatoren zur Anwendung kommen dürfen.

'__STDC__' ist im ANSI-C vordefiniert und ist TRUE, d.h. 1, wenn der Compiler alle Anforderungen des Standards unterstützt. Man kann so problemlos Programme für beide Arten von Übersetzern schreiben, indem man dieses neue Kriterium abfragt. Wurden in den meisten K&R-Implementationen zwei Definitionen eines Makros kommentarlos übergangen und die letzte als gültig erklärt, so sind beim ANSI-C Mehrfachdefinitionen ausdrücklich nicht erlaubt und mit einer Fehlermeldung abzubrechen. Auch die rekursiven Makros, die in einigen Programmen zu finden sind, sind nicht mehr erlaubt.

La langue même

In die Sprache selbst wurden einige neue Elemente aufgenommen, die von C++ stammen. Eines davon, die Prototypen, benutzt der Compiler, um auch bei der Aufteilung eines Programmes in mehrere Module die Funktionsparameter auf ihre Richtigkeit hin überprüfen zu können. Die meisten Typenprüfprogramme (LINT) konnten dies nur innerhalb eines Moduls erledigen. Diese Überprüfung reduziert Fehler bei der Parameterübergabe erheblich. Die Prototypen sollten am Anfang eines jeden Programmes oder in einer entsprechenden Header-Datei stehen.

```
void   load(char *name, int size);
```

ist ein Beispiel, wie ein Prototyp auszusehen hat. Er deklariert eine Funktion, die kein Ergebnis liefert, und einen String und einen Integer als Argumente erwartet. Die Namen der einzelnen Parameter können auch weggelassen werden. Die Prototypen sind gleichzeitig auch Funktionsdeklarationen. Nach dem selben Prinzip kann man auch Funktionen definieren. Eine Definition im neuen Stil sieht so aus

```
float  gamma(float real)
{
       ...
}
```

im Gegensatz zur alten

```
float  gamma(real)
float  real;
{
       ...
}
```

Man spart also auch an wertvoller Tipparbeit. Die alten Definitionen sind aus Kompatibilitätsgründen noch immer erlaubt und bringen keine Nachteile mit sich. Die Implementierung der variablen Anzahl der Parameter, wie sie zum Beispiel in allen printf-Funktionen verwendet wird, wurde im K&R den Herstellern überlassen. Im ANSI-C weisen drei Punkte '...' als Funktionsparameter den Compiler an, die nachfolgenden Argumente an die Funktion zu übergeben. Ein Beispiel

```
int    printf(char *format, ...);
```

Der Programmierer kann dann mit Hilfe dreier Makros an die Parameter gelangen. 'va_start(ap, parmN)' initialisiert ap, der als 'va_list' deklariert sein muß. 'parmN' ist das letzte 'feste' Argument. Mit 'va_arg(ap, type)' kommt man dann an einzelne Parameter, deren Typ anstelle von 'type' stehen muß. Das Ganze wird mit 'va_end(ap)' abgeschlossen. Im Beispiel von 'printf' liefert Ihnen argument

```
va_list arg_pointer;
int    argument;
va_start(arg_pointer, format);
argument = va_arg(arg_pointer, int);
va_end(arg_pointer);
```

den ersten variablen Wert. Funktionen mit variablen Argumenten müssen in Turbo C immer einen Prototyp haben, nur so erkennt der Compiler, daß er die Parameter über den Stack weitergeben soll. Sollten Sie mal vergessen, 'stdio.h' zu 'includen', wundern Sie sich nicht über die unvorgesehene Funktionsweise ihres Rechners, wenn Sie eine der printf-Funktionen ansprechen. In der Turbo-C-Dokumentation wird leider nicht erwähnt, daß Sie auf keinen Fall 'char' als Typ bei 'va_arg' verwenden dürfen: ANSI-gemäß werden nämlich 'chars' automatisch in 'ints' umgewandelt.

Auch auf der Sprachebene wurden neue Schlüsselwörter eingeführt. Da auf verschiedenen Computern Datentypen verschieden behandelt werden, kann es passieren, daß ein Typ mal vorzeichenbehaftet ist, mal nicht. Um den dadurch entstandenen Fehlern vorzubeugen, hat man als Gegenstück zu 'unsigned' 'signed' eingeführt. So kann der Programmierer unabhängig von Default-Einstellungen seine Typen festlegen. Die in vielen C-Implementationen bereits vorhandenen 'enum' und 'void' wurden als nützlich erkannt und in den Standard aufgenommen. Auch das altbekannte 'structure passing' ist im ANSI-C erlaubt. Die Weiterführung des Typs 'void' ergab 'void *', einen Pointer auf irgendetwas. Dies erleichtert vor allem das Handling mit den Speicherverwaltungsfunktionen.

Aus

```
char   *malloc(unsigned int size);
```

wird

```
void   *malloc(size_t size);
```

Man spart sich damit die explizite Typanpassung (type casting). Man muß aber auch beachten, daß mit void-Zeigern nicht einfach gerechnet werden kann; schließlich ist die Größe des Datentyps noch gar nicht bekannt. Zwei neue Typen 'const' und 'volatile' wurden eingeführt, um dem Optimierer die Arbeit zu erleichtern, denn sie zeigen, welche Variablen grundsätzlich konstant und welche flüchtig sind.

```
const  float e = 2.71828182;
```

initialisiert eine Konstante, die problemlos aus Schleifen gezogen werden kann. Der Übersetzer verbietet jede Änderung dieser Variable.

```
volatile time_t timer;
```

hingegen sagt dem Compiler, daß sich die Variable 'timer' von außen her, also durch Interrupts oder ähnliche nicht kontrollierbare Ereignisse verändert. Solche Variablen werden dann immer von neuem ausgelesen, damit der Wert auch der richtige ist. Wichtig ist dieser Typ vor allem in einer Multitasking-Umgebung. Auf Pointer angewandt, erlaubt 'const' noch eine kleine Feinheit: Es lassen sich sowohl die Daten, auf die sie zeigen,

```
const  char *name;
```

als auch die Zeiger selbst

```
char   * const name;
```

gegen Veränderungen schützen. Wenn Sie eigene Bibliotheken schreiben, dann sollten Sie mit diesen beiden Pointern arbeiten. Somit schließen Sie nämlich aus, daß Sie Werte verändern, die Sie nicht verändern dürfen.

'long double' ist ein weiterer neuer Datentyp, der eingeführt wurde. Er muß mindestens so genau sein wie 'double' und eignet sich gut für 10-Byte-Formate. Der Typ 'float' wurde im K&R auf 'double' erweitert, wenn er als Argument einer Funktion zugewiesen wurde. Dies ist nicht mehr der Fall.

Damit Kompatibilitätsprobleme vermieden werden, liefert der 'sizeof'-Operator 'size_t' (unsigned long). Der Compiler wird bei einer Portierung hier oft Warnungen ausgeben, da alte Programme meist Integer erwarten.

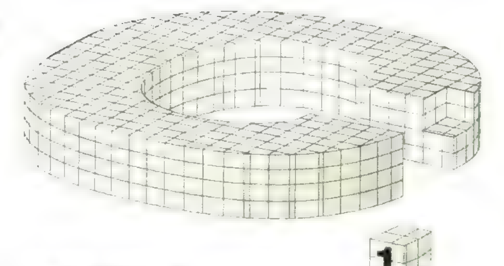
Das waren die wichtigsten Verbesserungen. Der Artikel nimmt nicht in Anspruch, vollständige ANSI-Referenz zu sein, das ist in diesem Umfang schließlich auch nicht möglich. Die Typenumwand-

lungen und Bibliotheksfunktionen habe ich aus- gelassen. Die zweite Ausgabe der 'C-Bibel' [1], die es auch in einer deutschen Fassung gibt, sei jedem empfohlen, der mit ANSI-C ernsthaft arbeiten möchte. Eine vollständige Referenz im Handbuchformat finden Sie in [2].

ST_Envy

Wenn Sie den ersten Teil des Artikels brav durchgelesen haben, verdienen Sie ein kleines Bonbon. Ich weiß, was Sie jetzt denken, und Sie haben recht. ST_Envy ist, wie der Name schon sagt, der kleine Bruder von Mac_Envy, einem hübschen Programm für den Macintosh, das Ihnen die Konfiguration Ihres Computers anzeigt. Da es auch von Atari verschiedene Computer gibt, die mit verschiedenen Betriebssystemen und Prozessoren ausgerüstet sind, entschloß ich mich, ST_Envy zu schreiben, damit Sie Ihre Freunde und Feinde beim Blick in die Innereien Ihres Computers neidisch machen können. Unten sehen Sie die zugegebenermaßen kurze Erklärung dieses Accessories, aber schließlich ist das Programm nicht als eine Studie zu betrachten.

Das Programm ist in GEM eingebunden und kann auch als Accessory gestartet werden. Ich habe bewußt auf Dialogboxen verzichtet und Ihnen damit die Erstellung einer Resource-Datei erspart. Stattdessen verwaltet ST_Envy ein kleines Fenster, welches Sie frei auf dem Desktop verschieben und schließen können. Hier zeigt sich ein kleiner Nachteil von AES und C: das Detail-Hiding fehlt. Der einfache Programmierer wird von den vielen Möglichkeiten erschlagen. Angebrachter wäre hier eine Strukturierung, die mit sehr leistungsfähigen Funktionen anfängt und dann stufenweise zu den low-level Routinen vordringt.



Nach erfolgreicher Initialisierung per 'gem_init' trägt sich unser Accessory in die Menüleiste ein und wartet, bis es angeklickt wird. Ähnlich geht es vor sich, wenn Sie ST_Envy als Programm starten, nur wird hier gleich unser Fenster geöffnet und dann erst auf die AES-Ereignisse gewartet. Das Öffnen erledigt die Funktion 'open_window'.

Nun sendet uns das AES die Nachricht, daß wir ein Fenster erneuern sollen. Dies erledigt 'do_redraw'. Diese Funktion finden Sie in jedem Programm, das Fenster verwaltet. Nachdem der neu zu zeichnende Bereich festgelegt und geclippt wurde, wird 'output' aufgerufen. Hier sehen Sie auch den Befehl 'perform(&setup)'. An dieser Stelle mußte ich einfach Assembler benutzen. Schauen Sie sich das Listing 2 an: ich habe es dokumentiert und möchte es nicht weiter erklären. Die Struktur 'setup' stellt die zum Abfragen nötigen Variablen bereit. Diese werden mit ihren Erklärungen nacheinander in die bereitgestelten Strings geschrieben. Ich habe dafür die Funktion 'sprintf' benutzt. Da in den Variablen selbst aber nur Zahlenwerte stehen, habe ich einige Felder, die mit Strings gefüllt sind, definiert und diese dann mit den Werten entspechender Variablen indiziert. So müssen Sie nicht hunderte von 'ifs' abschreiben. Trotzdem gestaltet sich die 'Inkarnation' der Zahlenwerte mit 13 Aufrufen von 'sprintf' etwas umständlich. Nun können die Zeichenfolgen über eine Schleife ins Fenster geschrieben werden. Die Funktion gibt die Strings einfach per 'v_gtext' relativ zu den Fensterkoordinaten aus. Sie muß dabei gar nichts beachten, weil der auszugebende Bereich bereits geclippt worden ist. Dann geht es wieder zurück zu 'events', wo in einer Schleife die AES-Funktion 'evnt_multi' aufgerufen wird.

Die Funktion 'do_message' reagiert auf die Ereignisse, die von GEM gemeldet werden. Wenn das Fenster geschlossen wird und ST_Envy als Programm gestartet worden ist, wird nach 'gem_exit' gesprungen und die Applikation verlassen.

Zum Compilieren gibt es nicht viel zu sagen. Sie müssen nur darauf achten, daß der Compiler die Variablen nicht über den Stack übergibt. Wenn Sie den Makro-Assembler von Borland verwenden, schalten Sie die Switches 'super', '68030' und '68881' ein(-s38). Ansonsten ersetzen Sie die Befehle, die Ihr Assembler nicht versteht, mit den entsprechenden '.dc.l'. Sie finden diese im Listing. Beim Linken reicht dem Accessory 1 Kilobyte Stack. Hier möchte ich nochmal dem Andreas danken, der mit seinem Wissen zu ST_Envy beigetragen hat.

So, das war's schon für diesmal. Ich hoffe, daß Ihnen die erste Folge unserer neuen Serie gefallen hat. Anregungen und Kritik senden Sie bitte an die Redaktion: diese wird sie dann an mich weiterleiten. Nächsten Monat geht es weiter mit der Speicherverwaltung.

Grischa Ekart

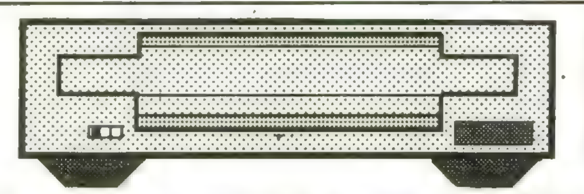
Literatur

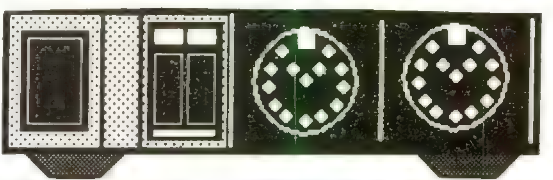
[1] B.W. Kernighan & D.M. Ritchie: The C Programming Language, Prentice-Hall, 1988

[2] P.J. Plauger & J. Brodie: Standard C, Microsoft Press, 1989

```
 1: /**************************************************/
 2: /*           ST_Envy V1.0D von 6.6.1990          */
 3: /* Autor: Grischa Ekart / (c) by MAXON Computer*/
 4: /*  erstellt mit Turbo C V1.0 von BORLAND GmbH */
 5: /**************************************************/
 6:
 7: #include <aes.h>
 8: #include <vdi.h>
 9: #include <stdio.h>
10: #include <stdlib.h>
11:
12: /*                  defines                      */
13:
14: #define TRUE 1
15: #define FALSE 0
16: #define min(x,y) (((x) < (y)) ? (x) : (y))
17: #define max(x,y) (((x) > (y)) ? (x) : (y))
18:
19: /*                 structures                    */
20:
21: struct WINDOW {
22:  int      handle;
23:  int      x, y, w, h;
24: } window;
25:
26: struct SETUP {
27:  int              computer;
28:  int              processor;
29:  int              coprocessor;
30:  int              blitter;
31:  size_t  mem, free_mem;
32:  int              hard_disk, hard_vers;
33:  int              dot_x, dot_y, colors;
34:  int              res_x, res_y;
35:  int              system, sys_vers;
36:  int              gdos;
37: } setup;
38:
39: /*              external definitions              */
40:
41: extern   int _app;
42:
43: /*               global definitions               */
44:
45: int      gl_apid;
46:
47: int      work_in[12];
48: int      work_out[57];
```

PDD DISC DRIVES 720, 950 KB + 1,2 u. 1,44 MB





PDD-SERIE

DIES IST PDD-3

Grundausstattung: 100% ST-Compatibel

- MIT NEC 1037a/TEAC
- MIT FLOPPYKABEL + NETZKABEL (> 1 m)
- INTEGR. SPEZIALNETZTEIL + TRAFO (INTERN)
- MIT STECKBAREN ANSCHLÜSSEN
- 2. BUCHSE, AN/AUS-SCHALTER
- 720 KB — 1,2 MB FORMATIERT
- 2. SCHALTER FÜR A/B VERTAUSCH

2 MB-Laufwerke 100 % ST-Kompatibel

NEU: 1,44 MB (formatiert)

genauso wie PDD-Serie, läuft an allen Atari wenn diese Atari's laut Umbauanleitung ST-Computer 1/90 auf 2 MB umgerüstet worden sind. Hard- und Softwarebeschreibung dazu: siehe ST-COMPUTER 1/90. An den Laufwerken muß nichts mehr verändert werden. Für 1 u. 2 MB.

Modell	Beschreibung	Preis
PDD-3	EINZELSTATION (720 KB)	335,—
PDD-13	DOPPELSTATION (2 x 720 KB)	585,—
PDD-5	5 1/4 Zoll, Umschalter vorne! 40/80 Tracks + Software, IBM-Compatibel	389,—
PDD-16	3 1/2 Zoll plus 5 1/4 Zoll in einem Gehäuse, wie PDD-3 plus PDD-5 zusammen	689,—
PDD-18	2 x 3 1/2" + 5 1/4 in einem Gehäuse A/B + 5/3 Schalter; wie PDD-5 + PDD-13 zusammen	898,—
PDD-4	Einzelstation (720 KB, 1,44 MB)	398,—
PDD-14	Doppelstation 2 x (720 KB, 1,44 MB)	698,—
PDD-7	5 1/4" (360 KB, 720 KB, 1,2 MB) 100% IBM, PC-Speed, Atari Kompatibel	398,—
PDD-20	3 1/2" + 5 1/4" in einem Gehäuse wie PDD-4 + PDD-7 zusammen	998,—
PDD-22	2 x 3 1/2" + 5 1/4", A/B + 5/3 Schalter; wie PDD-7 + PDD-14 zusammen	1.298,—

WEITERE FLOPPYSTATIONEN (EINF. AUSFÜHRUNG) NEC 1037 A grau 179,—

ST-3	ST-13	ST-5	ST-4	Floppy-Umbau
EINZELSTATION TEAC NEC 1037 KOMPL. ANSCHLUSSF. 199,— / 239,—	DOPPELSTATION INTEGRIERTES NETZT. 2 x 720 KB 399,—	5 1/4 ZOLL, TEAK 55FR wesentlich verbesserte Version 40/80 TRACK 299,—	Einzelstation TEAC 1,44 MB + 1 MB 299,—	Wir bauen Ihr SF 354 um, mit NEC 1037 A für nur 199,— DM doppelseitig

PC-Speed

Sofort lieferbar!

IBM-XT Emulator neueste Version

SOFORT LIEFERBAR **Preis 398,-**

SCSI-I

NEU: MHD-81, besser als MHD-80, extrem leise, 16 ms schnell, 5 x höhere Lebensdauer! **DM 1.998,-** **SUPER LEISE**

Professionelle SCSI-Platten für Atari-ST komplett anschlußfertig.
Alle Stationen haben integriertes Netzteil, neuester Treiber-Interface, 100% Atari-kompatibel. DMA steckbar, gepuffert SCSI (Option) Metallgehäuse grau (Mega ST-Abmessungen) Echtzeituhr + Software dabei.

Modell	Beschreibung	Preis
MHD-50:	50 MB Festplatte (28 ms Geräusch-leise)	1.398,—
MHD-80:	80 MB Festplatte 80 MB 2 PWS	1.898,—
MHD-150:	150 MB Festplatte	2.498,—

Alle Festplatten sind extrem leise u. extrem schnell. Alle Stationen haben: 28-40 ms. Die Laufwerke laufen mit einem Hochgeschwindigkeits Host-Adapter ungefähr 50% schneller als ältere Modelle. 100% ST-kompatibel: Sie können auch Fremddisk od. Atari-Harddisk anschließen. PC-Speed, Aladin, PC-Ditto, sonstige Emulatoren: keine Probleme. DMA Port: DMA Port komplett herausgeführt und gepuffert.
Software-Harddisk-Utilities: jeder Harddisk liegt ein professionelles Harddisk-Utility bei. D.h.: Hardware und Software kompatibel zum neuen Atari-Standard. Tos 1, — Tos 1.6 kompatibel; beliebig große Partionen, 14 weitere Partionen max., und weitere Besonderheiten.

Mega ST-Einbaukit: Unsere Festplatten passen in Ihren Rechner.

Weitere Infos auf Anfrage.

DIE ALTERNATIVE ZUR ATARI WECHSELPLATTE

DISC-file-20

100% Atari kompatibel — 20 MB auf einer 5 1/4" Diskette — Zugriffszeit: 65 m/sec. — laufwerk mit einer Diskette und Treibersoftware — Diskettenpreis 80,— / 60 Stück

MS-DOS-Version in Vorbereitung **DM 1.998,-**

FISCHER-COMPUTER

Goethestraße 7 — 6101 Fränkisch-Crumbach

Telefon 06164-4601 + 3748 · abends zwischen 18.00 + 22.00 Uhr ebenso · Telefax 06164-3748

```
 49:
 50:  int       handle,
 51:            phys_handle;
 52:
 53:  int       hidden;
 54:
 55:  int       gl_hchar,
 56:            gl_wchar,
 57:            gl_hbox,
 58:            gl_wbox;
 59:
 60:  int       max_x,
 61:            max_y;
 62:
 63:  int       menu_id;
 64:
 65:  /*              internal prototypes              */
 66:
 67:  void      main(void);
 68:  void      gem_init(void);
 69:  void      gem_exit(void);
 70:  void      events(void);
 71:  int       do_message(int msg_buf[8]);
 72:  void      open_window(void);
 73:  void      close_window(void);
 74:  void      do_redraw(void);
 75:  int       rc_intersect(GRECT *t1, GRECT *t2);
 76:  void      show_mouse(void);
 77:  void      hide_mouse(void);
 78:  void      set_window(int x, int y, int w, int h);
 79:  void      output(void);
 80:
 81:  /*              external prototypes              */
 82:
 83:  void      perform(struct SETUP *setup);
 84:
 85:  /*              function definitions             */
 86:
 87:  void
 88:  main(void)
 89:  {
 90:   gem_init();
 91:   if(!_app)
 92:   {
 93:            menu_id = menu_register(gl_apid,
 94:                    "  ST_Envy");
 95:   }
 96:   else
 97:   {
 98:            graf_mouse(ARROW, NULL);
 99:            open_window();
100:   }
101:   events();
102:   gem_exit();
103:  }
104:
105:  void
106:  gem_init(void)
107:  {
108:   int i;
109:
110:   if((gl_apid = appl_init()) == -1)
111:            exit(-1);
112:
113:   for (i = 0; i < 10; work_in[i++] = 0)
114:            ;
115:
116:   work_in[10] = 2;
117:   phys_handle = graf_handle(&gl_wchar, &gl_hchar,
118:                 &gl_wbox, &gl_hbox);
119:
120:   work_in[0] = handle = phys_handle;
121:   v_opnvwk(work_in, &handle, work_out);
122:   if(handle == FALSE)
123:   {
124:            appl_exit();
125:            exit(-2);
126:   }
127:
128:   setup.dot_x = max_x = work_out[0] + 1;
129:   setup.dot_y = max_y = work_out[1] + 1;
130:   setup.colors = work_out[13];
131:   setup.res_x = work_out[3];
132:   setup.res_y = work_out[4];
133:   if(vq_gdos())
134:            setup.gdos = TRUE;
135:   else
136:            setup.gdos = FALSE;
137:
138:   window.x = 100;
139:   window.y = 100;
140:   window.w = 188;
141:   window.h = 133;
142:   window.handle = FALSE;
143:  }
144:
145:  void
146:  gem_exit(void)
147:  {
148:   v_clsvwk(handle);
149:   appl_exit();
150:   exit(0);
151:  }
152:
153:  void
154:  events(void)
155:  {
156:   int      quit = FALSE;
157:   int      msg_buf[8];
158:   int      x, y,
159:                    kstate,
160:                    key,
161:                    clicks,
162:                    event,
163:                    state;
164:
165:   do
166:   {
167:            event = evnt_multi(MU_MESAG,
168:                    2, 1, 1,
169:                    0, 0, 0, 0, 0,
170:                    0, 0, 0, 0, 0,
171:                    msg_buf,
172:                    0, 0,
173:                    &x, &y, &state, &kstate,
174:                    &key, &clicks);
175:
176:            wind_update(BEG_UPDATE);
177:            if(event & MU_MESAG)
178:                    quit = do_message(msg_buf);
179:
180:            wind_update(END_UPDATE);
181:   } while(!quit);
182:  }
183:
184:  int
185:  do_message(int msg_buf[8])
186:  {
187:   switch(msg_buf[0])
188:   {
189:            case AC_OPEN:
190:                    if(msg_buf[4] == menu_id)
191:                            open_window();
192:            break;
193:            case AC_CLOSE:
194:                    if(msg_buf[3] == menu_id)
195:                            window.handle = FALSE;
196:            break;
197:            case WM_REDRAW:
198:                    if(msg_buf[3] == window.handle)
199:                            do_redraw();
200:            break;
201:            case WM_NEWTOP:
202:            case WM_TOPPED:
203:                    wind_set(window.handle, WF_TOP);
204:            break;
205:            case WM_CLOSED:
206:                    if(msg_buf[3] == window.handle)
207:                            close_window();
208:                    if(_app)
209:                            return(TRUE);
210:            break;
211:            case WM_MOVED:
212:                    if(msg_buf[3] == window.handle)
213:                            set_window(msg_buf[4],
214:                                    msg_buf[5],
215:                                    msg_buf[6],
216:                                    msg_buf[7]);
217:            break;
218:   }
219:   return(FALSE);
220:  }
```

→

```
221:
222:  void
223:  open_window(void)
224:  {
225:   if(window.handle == FALSE)
226:   {
227:          window.handle = wind_create(
228:                 NAME | CLOSER | MOVER,
229:                 0, 0, max_x, max_y);
230:
231:          if(window.handle <= 0)
232:          {
233:                 form_alert(1,
234:                    "[1][Keine weiteren Fenster|"
235:                    "vorhanden! Räumen Sie die|"
236:                    "geöffneten Fenster ab.]"
237:                    "[ Bis später ]");
238:
239:                 window.handle = FALSE;
240:                 return;
241:          }
242:          wind_set(window.handle,WF_NAME,
                    "ST_ENVY");
243:          wind_open(window.handle, window.x,
                     window.y,
244:                  window.w, window.h);
245:   }
246:   else
247:          wind_set(window.handle, WF_TOP);
248:  }
249:
250:  void
251:  close_window(void)
252:  {
253:   wind_close(window.handle);
254:   wind_delete(window.handle);
255:   window.handle = FALSE;
256:  }
257:
258:  void
259:  set_window(int x, int y, int w, int h)
260:  {
261:   wind_set(window.handle, WF_CURRXYWH,
262:          x ,y ,w, h);
263:
264:   window.x = x;
265:   window.y = y;
266:   window.w = w;
267:   window.h = h;
268:  }
269:
270:  void
271:  do_redraw(void)
272:  {
273:   int      pxyarray[4];
274:
275:   GRECT    box,
276:                 work;
277:
278:
279:   if(window.handle <= 0)
280:          return;
281:
282:   hide_mouse();
283:
284:   wind_get(window.handle, WF_WORKXYWH, &work.g_x,
285:          &work.g_y, &work.g_w, &work.g_h);
286:
287:   wind_get(window.handle, WF_FIRSTXYWH, &box.g_x,
288:          &box.g_y, &box.g_w, &box.g_h );
289:
290:   work.g_w = min(work.g_w, max_x - work.g_x);
291:   work.g_h = min(work.g_h, max_y - work.g_y);
292:
293:   while(box.g_w && box.g_h)
294:   {
295:          if(rc_intersect(&work, &box))
296:          {
297:                 pxyarray[0] = box.g_x;
298:                 pxyarray[1] = box.g_y;
299:                 pxyarray[2] = box.g_x+box.g_w-1;
300:                 pxyarray[3] = box.g_y+box.g_h-1;
301:
302:                 vs_clip(handle, TRUE, pxyarray);
303:                 vr_recfl(handle, pxyarray);
304:                 output();
305:          }
306:          wind_get(window.handle, WF_NEXTXYWH,
307:          &box.g_x, &box.g_y, &box.g_w, &box.g_h);
308:   }
309:   vs_clip(handle, FALSE, pxyarray);
310:   show_mouse();
311:  }
312:
313:  int
314:  rc_intersect(GRECT *t1, GRECT *t2)
315:  {
316:   int      xmin,
317:                 xmax,
318:                 ymin,
319:                 ymax;
320:
321:   xmax = min(t2->g_x+t2->g_w, t1->g_x+t1->g_w);
322:   ymax = min(t2->g_y+t2->g_h, t1->g_y+t1->g_h);
323:   xmin = max(t1->g_x, t2->g_x);
324:   ymin = max(t1->g_y, t2->g_y);
325:
326:   t2->g_x = xmin;
327:   t2->g_y = ymin;
328:   t2->g_w = xmax - xmin;
329:   t2->g_h = ymax - ymin;
330:
331:   return(t2->g_w > 0 && t2->g_h > 0);
332:  }
333:
334:  void
335:  show_mouse(void)
336:  {
337:   if(hidden)
338:   {
339:          graf_mouse(M_ON, NULL);
340:          hidden = FALSE;
341:   }
342:  }
343:
344:  void
345:  hide_mouse(void)
346:  {
347:   if(!hidden)
348:   {
349:          graf_mouse(M_OFF, NULL);
350:          hidden = TRUE;
351:   }
352:  }
353:
354:  void
355:  output(void)
356:  {
357:   int      i;
358:
359:   GRECT    work;
360:
361:   char     temp_str[35];
362:   char     string[12][35];
363:   char     *list[] = {
364:            "COMPUTER:     %s",
365:            "PROZESSOR:    MOTOROLA %s",
366:            "COPROZESSOR:  %s",
367:            "BLITTER:      %s",
368:            "SPEICHER:     %luK, %luK FREI",
369:            "FESTPLATTE:   %s",
370:            "TREIBER:      %s",
371:            "BILDSCHIRM:   %d x %d DOTS",
372:            "FARBEN:       %d",
373:            "AUFLÖSUNG:    %d x %d DPI",
374:            "SYSTEM:       %s-TOS V%.1f",
375:            "GDOS:         %s"
376:   };
377:   char     *computer[] = {
378:            "ATARI ST",
379:            "ATARI STE",
380:            "ATARI TT"
381:   };
382:   char     *processor[] = {
383:            "68000",
384:            "68010",
385:            "68020",
386:            "68030"
387:   };
388:   char     *installed[] = {
389:            "NICHT INSTALLIERT",
390:            "INSTALLIERT"
```

→

```
391:    };
392:    char    *tos[] = {
393:            "RAM",
394:            "ROM"
395:    };
396:
397:    perform(&setup);
398:    vst_color(handle, 1);
399:    vst_height(handle, 4, &gl_wchar, &gl_hchar,
400:            &gl_wbox, &gl_hbox);
401:
402:    wind_get(window.handle, WF_WORKXYWH, &work.g_x,
403:            &work.g_y, &work.g_w, &work.g_h);
404:
405:    work.g_x += 4;
406:    gl_hbox += 3;
407:
408:    sprintf(string[0], list[0],
409:            computer[setup.computer]);
410:
411:    sprintf(string[1], list[1],
412:            processor[setup.processor]);
413:
414:    sprintf(string[2], list[2],
415:            installed[setup.coprocessor]);
416:
417:    sprintf(string[3], list[3],
418:            installed[setup.blitter]);
419:
420:    sprintf(string[4], list[4],
421:            setup.mem, setup.free_mem);
422:
423:    sprintf(string[5], list[5],
424:            installed[setup.hard_disk]);
425:
426:    if(setup.hard_disk)
427:    {
428:            if(setup.hard_vers)
429:            {
430:                    sprintf(temp_str, "AHDI V%.2f",
431:                      (float)setup.hard_vers / 100);
432:
433:              sprintf(string[6], list[6], temp_str);
434:            }
435:            else
436:                    sprintf(string[6], list[6],
437:                            installed[TRUE]);
438:    }
439:    else
440:            sprintf(string[6], list[6],
441:                    installed[FALSE]);
442:
443:    sprintf(string[7], list[7],
444:            setup.dot_x, setup.dot_y);
445:
446:    sprintf(string[8], list[8], setup.colors);
447:    sprintf(string[9], list[9],
448:          25400 / setup.res_x, 25400 / setup.res_y);
449:
450:    sprintf(string[10], list[10],
451:            tos[setup.system],
452:            (float)setup.sys_vers / 10);
453:
454:    sprintf(string[11], list[11],
455:            installed[setup.gdos]);
456:
457:    for(i = 0; i < 12; i++)
458:            v_gtext(handle, work.g_x,
459:          work.g_y + (i + 1) * gl_hbox, string[i]);
460:    }
```

```
 1: ;          Perform V1.0 10.6.1990
 2: ;          (w) & (c) by Grischa Ekart
 3:
 4:  .globl  perform
 5:
 6:  .text
 7:
 8: perform:
 9:  movem.l d0-a6, -(sp)     ;rette alles
10:  move.l  a0, a6           ;&setup nach a6
11:  clr.l           -(sp)
12:  move            #$20, -(sp)
13:  trap            #1       ;supervisor mode
14:  addq.l  #6, sp
15:  move.l  d0, ssp_save     ;rette ssp
16:  move.l  $42e, d0         ;Speicher
17:  divu.w  #$400, d0        ;in Kilobyte,
18:  ext.l           d0       ;in longs
19:  move.l  d0, 8(a6)        ;nach setup->mem.
20:  move.l  #-1, -(sp)
21:  move            #$48, -(sp)
22:  trap            #1       ;freier Speicher
23:  addq.l  #6, sp
24:  divu.w  #$400, d0        ;in Kilobyte,
25:  ext.l           d0       ;in longs
26:  move.l  d0, 12(a6)       ;setup->free_mem.
27:  moveq           #0, d0  ;treiber
28:  move            d0, 18(a6)  ;nicht installiert
29:  move.l  $516, a0            ;&pun_info
30:  cmp.l           d0, a0
31:  beq             weiter  ;kein Treiber
32:  tst             (a0)
33:  beq             weiter  ;keine Festplatte
34:  move            #1, 16(a6) ;Festplatte
35:  lea             82(a0), a0 ;&P_cookie
36:  cmp.l           #$41484449, (a0)  ;AHDI?
37:  bne             weiter    ;nein
38:  cmp.l           4(a0), a0  ;P_cookptr
39:  bne             weiter    ;!= &P_cookie
40:  move.b  8(a0), d0        ;Versionsnr.
41:  ext.w           d0       ;aufbereiten
42:  mulu            #100, d0
43:  move.b  9(a0), d1
44:  move.b  d1, d2
45:  andi.b  #$f, d1
46:  andi.b  #$f0, d2
47:  ext.w           d1
48:  ext.w           d2
49:  lsr             #8, d2
50:  mulu            #10, d2
51:  add             d1, d0
52:  add             d2, d0
53:  move            d0, 18(a6)  ;Versionsnr.* 100
54: weiter:
55:  move.l  $4f2, a0            ;_sysbase
56:  move            #0, 30(a6)  ;RAM=>system
57:  cmp.l           $436, a0    ;_memtop
58:  blt             ram         ;kleiner?
59:  move            #1, 30(a6)  ;ROM=>system
60: ram:
61:  move.b  2(a0), d0           ;Versionsnr.
62:  ext.w           d0          ;aufbereiten
63:  mulu            #10, d0
64:  move.b  3(a0), d1
65:  ext.w           d1
66:  add             d1, d0
67:  move            d0, 32(a6)  ;Versionsnr.* 10
68:  move.l  a7, a5              ;rette a7
69:  move.l  $8, bus_save        ;rette Vektoren
70:  move.l  $10, ill_save
71:  move.l  $20, priv_save
72:  move.l  $2c, lf_save
73:  move.l  #bus_handle, $8     ;eigene
74:  move.l  #ill_handle, $10
75:  move.l  #priv_handle, $20
76:  move.l  #ill_handle, $2c
77:  move            #$700, sr   ;user mode
78:  lea             weiter1, a4 ;Rücksprung
79:  move            sr, d0
80: ;dieser Befehl löst bei allen CPUs auPer
81: ;68000 eine Privilegverletzung aus
82:  moveq           #0, d0      ;CPU = 68000
83: weiter1:
84: ;d0 enthält jetzt die CPU
85:  move            d0, 2(a6)   ;=>processor
86:  lea             super, a4   ;Rücksprung
87:  move            0, d0       ;Buserror
88: ;wieder zurück in den supervisor mode
89: super:
90:  lea             weiter2, a4 ;Rücksprung
91:  move            $fffffa40, d0   ;FPU
92: ;beim 68000 ST ist die FPU in diesem
93: ;Adressbereich. Ist sie nicht installiert
94: ;gibt es einen Buserror
```

→

```
 95:   moveq          #1, d0       ;installiert
 96:  weiter2:
 97:   tst            d0           ;installiert?
 98:   bne            _68881       ;ja
 99:   move.l  #bus_handle, $2c    ;eigene Line-F
100:   moveq          #1, d0
101:   lea            _68881, a4   ;Rücksprung
102:  ;ist ein anderer Prozessor als 68000
103:  ;eingebaut, dann schickt er alle Befehle
104:  ;für die FPU automatisch an die FPU
105:  ;der folgende Befehl löst Line-F aus
106:  ;wenn keine FPU eingebaut ist
107:   fnop
108:  ;oder .dc.l $f2800000
109:  _68881:
110:   move           d0, 4(a6)    ;=>coprocessor
111:   lea            weiter3, a4  ;Rücksprung
112:   move           $ffff8a00, d0   ;Blitter?
113:  ;Buserror wenn kein Blitter
114:   moveq          #1, d0       ;ja
115:  weiter3:
116:   move           d0, 6(a6)    ;=>blitter
117:   cmp            #3, 2(a6)    ;68030?
118:   beq            atari_tt     ;ja
119:   lea            atari_st, a4  ;Rücksprung
120:   move           $e00000, d0  ;STE-ROM?
121:   bra            atari_ste    ;ja
122:  atari_st:
123:   move           #0, (a6)     ;=>computer
124:   bra            weiter4
125:  atari_ste:
126:   move           #1, (a6)     ;=>computer
127:   bra            weiter4
128:  atari_tt:
129:   move           #2, (a6)     ;=>computer
130:  weiter4:
131:   move           #$2000, sr   ;Interrupts frei
132:   move.l  a5, a7              ;alter Stack
133:   move.l  bus_save, $8        ;alte Vektoren
134:   move.l  ill_save, $10
135:   move.l  priv_save, $20
```

```
136:   move.l  lf_save, $2c
137:   move.l  ssp_save, -(sp)
138:   move           #$20, -(sp)
139:   trap           #1           ;back to life
140:   addq.l  #6, sp
141:   movem.l (sp)+, d0-a6        ;Register
142:   rts                         ;home
143:
144:  priv_handle:
145:   moveq          #1, d0       ;68010
146:   movec          caar, d1
147:  ;oder .dc.l $4e7a1802
148:  ;der Cache-Befehl nur >68010
149:   moveq          #2, d0       ;68020
150:   pmove          tt0, d1
151:  ;oder .dc.l $f0016a00
152:  ;pmmu nur ab 68030
153:   moveq          #3, d0       ;68030
154:   jmp            (a4)         ;zurück
155:
156:  bus_handle:
157:   moveq          #0, d0       ;nicht installiert
158:   jmp            (a4)         ;zurück
159:
160:  ill_handle:
161:   jmp            (a4)         ;zurück
162:
163:   .bss
164:
165:  ssp_save:
166:   ds.l           1
167:  bus_save:
168:   ds.l           1
169:  ill_save:
170:   ds.l           1
171:  priv_save:
172:   ds.l           1
173:  lf_save:
174:   ds.l           1
175:
176:   .end
```

METAMORPHOSE

VOM MIDIPORT ZUR SERIELLEN SCHNITTSTELLE

Sicher hat sich schon mancher ST-Anwender geärgert, daß sein ST nur eine RS-232-Schnittstelle hat - der TT dagegen gleich vier davon. Doch deshalb brauchen Sie nicht gleich auf den TT umzusteigen. In Verbindung mit dem MIDI-ACIA läßt sich am ST relativ einfach eine zweite serielle Schnittstelle (COM2) realisieren. Zwar muß man dabei in Kauf nehmen, daß der MIDI-Port nicht angesprochen werden kann, solange man die zweite RS-232 benutzt, aber wer spielt schon auf seinem Keyboard und macht zur gleichen Zeit DFÜ?

Bild 1: Der Schaltplan für die zweite serielle Schnittstelle

COM2 am ST

Das MIDI-ACIA, welches das Herz unserer Schnittstelle darstellen wird, ist eigentlich schon eine serielle Schnittstelle, nur mit der Eigenschaft, daß die Signale nicht im RS-232C-Format, sondern als TTL-Pegel anliegen, und daß zwei wichtige Steuerleitungen nicht vorhanden bzw. unbrauchbar sind.

Damit wir die Pegel zwischen beiden Normen umwandeln können, müssen wir als erstes die Festlegung der RS-232-Spannungen nach den CCITT-Normen V.24 und V.28 wissen:

logisch H = -3V bis -15V
logisch L = +3V bis +15V

Die Pegelumwandlung übernehmen zwei spezielle Treiberbausteine, welche auch an der normalen RS-232 (COM1) im ST eingesetzt sind. Für diese Aufgabe braucht der Sendetreiber (MC 1488) die Spannungen +12V und -12V. Da -12V vom ST-Netzteil nicht erzeugt wird, bedient man sich am besten an den schon im ST vorhandenen MC 1488. Man lötet einfach die Stomversorgungsanschlüsse des zweiten huckepack darauf und biegt die restlichen Pins nach oben. Auf die gleiche Weise wird auch der Empfangstreiber MC 1489 auf den vorhandenen 1489 gesetzt.

Jetzt lötet man entweder das ganze ACIA aus und sokkelt es, oder die Pins für Sende- und Empfangsdaten (RxD, TxD) am ACIA werden mit etwas Geschick duchgezwickt und hochgebogen. Beide Datenleitungen werden dann auf die Schalter S1 und S2 gelegt und für die Schalterstellung "MIDI" wieder auf die Platine zurückgeführt (siehe Schaltplan). Für die Schalterstellung "COM2" laufen die Signale durch die RS-232-Treiber und liegen an der Sub-D25-Buchse (CN1) an.

Zwar könnte man jetzt schon Daten auf unserer Schnittstelle ausgeben, allerdings nur mit der festeingestellten MIDI-Baud-Rate von 31.250 Baud. Weil diese für den RS-232-Betrieb unnütz ist, greifen wir für die Baud-Rate zu einem Trick, um auf brauchbare Raten zu kommen: Wir sorgen dafür, daß man die Frequenz des Sende- und Empfangstakts verändern kann.

Die ACIA-Pins für Sende- und Empfangstakt (RxCLK und TxCLK) werden

auch von der Platine getrennt und schaltbar (S4 und S5) mit dem Timer-A- (TAO des MFP), dem Timer-D-Ausgang (TDO) und dem normalen 500 kHz-Takt von der Platine verbunden. TxCLK und RxCLK können dabei zusammen auf eine Leitung gelegt werden, denn Sende- und Empfangs-Baud-Rate sind immer gleich (außer bei BTX...). Um Störungen bei schnellen Übertragungen zu vermeiden, ist es ratsam, möglichst kurze, am besten sogar Koax-Leitungen zu verwenden. Am Taktschalter sind nun folgende Einstellungen möglich:

S4 in Stellung "MIDI": 500 kHz - normale MIDI-Baud-Rate (fest)
S4 in Stellung "COM2": abhängig von S5: Timer-A oder Timer-D
Timer-A: Geschwindigkeit ist frei programmierbar
Timer-D: gleiche Baud-Rate wie die erste RS-232

METAMORPHOSE

VOM MIDIPORT ZUR SERIELLEN SCHNITTSTELLE

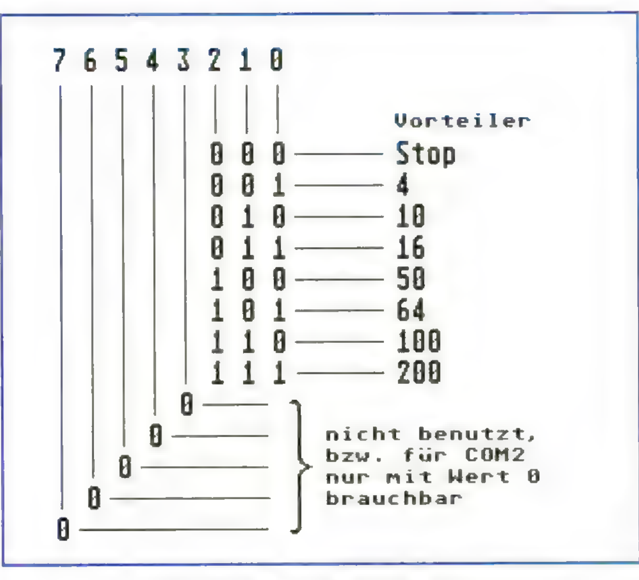


Tabelle 1: Timer A Control Register

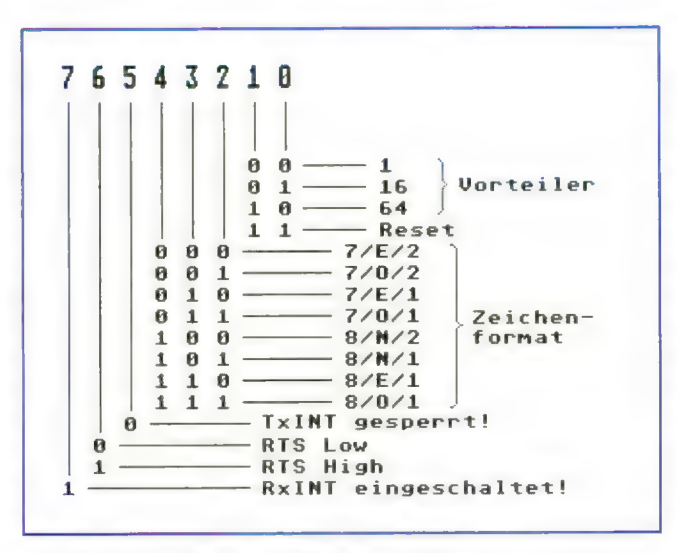


Tabelle 2: ACIA Steuerregister

Nun sind die beiden Datenleitungen voll funktionsfähig, doch zu einer echten RS232-Schnittstelle gehören noch einige Steuerleitungen, die wir alle (bis auf Ring Indicator) implementieren wollen. Die Signale werden dabei genau wie bei RxD und TxD mit Hilfe der Treiber zwischen RS232- und TTL-Pegel gewandelt.

RTS (Request to Send) und CTS (Clear to Send) sind die sogenannten Handshake-Leitungen. Beide kann man sich vom ACIA besorgen, welches diese bisher nicht nutzt, da es bei der Datenübertragung per MIDI keinen Hardware-Handshake gibt.

Doch wie funktioniert der RTS/CTS-Handshake an der RS232? Indem man das RTS-Signal auf log. L setzt, kann man dem angeschlossenen Modem (o.ä.) mitteilen, daß es den Empfangsdatenstrom anhalten soll. Bei log. H geht das Modem davon aus, daß der Computer zum Empfang bereit ist.

Mit der CTS-Leitung "sagt" das Modem dem Computer, ob er Daten senden darf oder nicht. Liegt diese auf log. H, ist das Modem derzeit nicht sendebereit, und das ACIA wartet mit dem Senden, bis sie wieder den Zustand L hat. Man sollte deshalb bei selbstgebastelten Nullmodem-Kabeln etc. darauf achten, daß RTS und CTS belegt sind. Ist dies nicht der Fall, hängt sich der ST beim Senden auf, da er vergeblich auf ein Clear to Send (log. L an CTS) wartet. Das ist auch der Grund, warum CTS normalerweise (MIDI-Betrieb) am ACIA auf Masse gelegt ist, denn das ACIA glaubt dann, daß es immer senden darf. Wenn Ihr Modem keinen RTS/CTS-Handshake unterstützt, ist es das beste, Sie verbinden RTS mit CTS (an CN1) durch einen Jumper - brauchen Sie doch einmal diesen Handshake, können Sie die Leitungen leicht wieder trennen.

Alles klar bis jetzt? Gut, dann gleich weiter mit dem Carrier-Detect-Signal (DCD). Wenn es auf log. H ist, weiß der Computer, daß die DÜE (Datenübertragungseinheit = Modem) mit einer Gegenstation "CONNECT"ed ist.

Wer nun glaubt, man kann dieses Signal über den stillgelegten DCD-Eingang des ACIAs abfragen, irrt sich, denn unser ACIA weigert sich, Daten zu senden, falls an diesem Eingang mit log. L signalisiert wird, daß das Modem keine Verbindung aufgebaut hat. Das würde bedeuten, daß man keine Befehle an das Modem schicken kann, solange es keinen Carrier empfängt, und so wäre z.B. automatisches Wählen ("ATD...") unmöglich.

Aber wie kommt man nun an ein vernünftiges Carrier-Detect-Signal? Die Lösung sieht folgendermaßen aus: Man speist das DCD von COM2 an dem MFP-Eingang (IO1) ein, an dem bereits das DCD-Signal der ersten RS232-Schnittstelle anliegt - nur eben nicht gleichzeitig, sondern umschaltbar! Diese Umschaltung erledigt der Multiplexer-Baustein 74HC4051 (IC3), der Ihnen bekannt sein dürfte, wenn Sie das HD-Laufwerksprojekt (nach [3]) in Ihren ST eingebaut haben. Vielleicht haben Sie ja damals gleich mehrere von den Multiplexern gekauft... Doch zurück zur DCD-Umschaltung: Man legt die DCD-Leitung von COM1 (MFP-Pin 23 in der Mitte abkneifen - siehe Schaltplan) und von COM2 auf zwei Eingänge des Multiplexers und den Ausgang zurück zum MFP-DCD-Eingang. Jetzt kann man mit der als Ausgang geschalteten Soundchip-Leitung IOA7, welche den A0-Eingang des Multiplexers bedient, die DCD-Leitung selektieren, die man am MFP abfragen will.

Als letztes Signal bleibt noch DTR übrig, welches sehr einfach zu realisieren ist. Das am PSG-Ausgang IOA6 anliegende Signal wird, nachdem es den Sendetreiber durchlaufen hat, auf Pin 20 am Sub-D25-Stecker angelegt.

Damit ist der hardware-mäßige Teil der Schnittstelle abgeschlossen, also wollen wir uns der Ansteuerung zuwenden. Grundsätzlich glaubt der ST immer noch, daß er es mit seiner normalen MIDI-Schnittstelle zu tun hat, und somit funktioniert die Ein- bzw. Ausgabe über das Ansprechen der MIDI-Schnittstelle (z.B. Bconin, Bconout). Die Baud-Rateneinstellung erfolgt für die Taktschalterstellung "Timer-D" über die XBIOS-Funktion Rsconf. Dieser Betriebssystemaufruf stellt die Ausgangsfrequenz des Timer-D so ein, daß die Standard-RS-232 die Baud-Rate hat, die man bei Rsconf als Parameter angegeben hat. Da sich das Midi-ACIA jetzt ebenfalls am Timer-D mit Takt ver-

METAMORPHOSE

VOM MIDIPORT ZUR SERIELLEN SCHNITTSTELLE

sorgt, hat die zweite serielle Schnittstelle immer die gleiche Geschwindigkeit wie die erste.

Um die Baud-Rate unabhängig von der ersten RS-232 einzustellen, gibt es eine weitere Möglichkeit, nämlich durch den Timer-A. An diesem MFP-Ausgang läßt sich eine fast beliebige Frequenz erzeugen, also genau das, was wir benötigen. Noch dazu wird er vom Betriebssystem nicht benutzt. Um ihn zu programmieren, muß man wissen, daß er über das "Timer-A-Data-Register" (Adresse $FFF A1F) und "Timer-A-Control-Regster" ($FFF A19) anzusprechen ist. Sobald man in das Control-Register einen Wert (siehe Tabelle) schreibt, wird ein Vorteiler aktiviert, der jeweils nach einer entsprechenden Anzahl von Taktzyklen (MFP-Takt = 2.4576 MHz) das Timer-Data-Register dekrementiert. Erreicht das Data-Register den Wert Null, bekommt es wieder den Anfangswert, und der Timer-Ausgang (TAO) wechselt seinen logischen Zustand. Dieser Vorgang wiederholt sich so lange automatisch, bis man ins Control-Register den Wert Null schreibt und damit den Timer stoppt. Die Frequenz, die sich durch den ständigen Zustandswechsel am Timer-Ausgang ergibt, dient dem ACIA dann als Sende- und Empfangstakt. Nachdem dieser noch die ACIA-interne Teilerstufe (Standardeinstellung: 1/16) durchlaufen hat, stellt er die Baud-Rate dar, welche sich folgendermaßen berechnen läßt:

$$\text{Baudrate} = \frac{2.4576\,\text{MHz} * \text{MFP-Timer-Vorteiler} * \text{ACIA-Vorteiler}}{2 * \text{Timer-Data-Startwert}}$$

Verwendet man den Standardvorteiler des ACIAs, erreicht man ebenso wie die erste serielle Schnittstelle eine maximale Baud-Rate von 19.2 kBaud. Mehr sieht die RS-232-Norm zwar nicht vor; aber wer sagt denn, daß wir nicht noch einen Zahn zulegen können? Indem man die beiden niederwertigsten Bits im ACIA-Steuerregister ($FFFC04) auf Null setzt, stellt man einen Frequenzteilerfaktor von 1:1 ein, und das bedeutet, daß man theoretisch eine Geschwindigkeit von bis zu 307.2 kBaud erreicht. Obwohl das ACIA laut [4] Baud-Raten von bis zu 500 kBaud verarbeiten kann, synchronisiert es beim Betrieb mit einem Vorteiler von Eins die Empfangsdaten nicht mehr mit der Taktfrequenz. Dies bedeutet, daß wir zwar auf den echten Highspeed-Bereich - also 100 kBaud und mehr - verzichten müssen, aber die, von manchen so heiß ersehnte, Baud-Rate von 38400 bps kann man allemal noch erreichen.

BAUTEILE:	
IC1:	MC1488
IC2:	MC1489
IC3:	74HC4051
C1, C2, C3:	10pF
S1 - S4:	4xUM-Schalter
S5:	1xUM-Schalter
CN1:	Sub-D25 Stecker

Tabelle 3: Diese preiswerten Bauteile werden benötigt.

Händeschütteln

Im Vergleich zu den komplizierten Vorteiler-Wursteleien für die Baud-Rate sind die Handshake-Leitungen ganz einfach zu bedienen: Indem man Bit 6 im ACIA-Steuerregister setzt, stellt man den Zustand des RTS-Anschlusses auf log. High ein. Der Zustand von CTS wird durch das Bit 3 im ACIA-Statusregister repräsentiert. Beide ACIA-Register liegen auf der gleichen Adresse ($FF FC 04), wobei auf das Statusregister durch Schreiben, auf das Steuerregister durch Lesen zugegriffen wird.

Das Ansprechen der DTR- bzw. DCD-Leitung ist in den Listings 1 und 2 dokumentiert. Nun ist eigentlich nur noch zu hoffen, daß möglichst viele Entwickler von RS-232-Anwendungen dem Beispiel Michael Bernards' folgen, denn dieser arbeitet bereits an einer Version seines Terminalprogrammes RUFUS (inzwischen bekannt und beliebt), welche die zweite serielle Schnittstelle voll unterstützen wird. Als weitere Anwendung für COM2 ist der Software-T-Switch (Listing 3) gedacht: Damit kann man alle BIOS-Ein- und Ausgaben von COM1 auf COM2 umlenken und so manche noch nicht angepaßten Programme überlisten. Es lohnt sich also auf jeden Fall, COM2 einzubauen, zumal alles zusammen kaum mehr als 10 Mark kostet.

A.Krieglmeier

Literatur:

[1] Jankowski, Rabich, Reschke: Atari ST Profibuch, Sybex 1988, ISBN 3-88745-563-0

[2] Brückmann, Englisch, Gerrits, Atari ST Intern, Data Becker 1987, ISBN 3-89011-119-X

[3] Bohs, Brod, Stepper: "Ganz schön dicht", ST-Computer 1/90

[4] "Asynchronous Communications Interface Adapter (ACIA), MC6850, Motorola Semiconductors

[5] Linear Integrated Circuits, "Computer and Terminal Interface", Motorola Semionductors

[6] Joe Campbell: V24 / RS-232 Kommunikation, Sybex 1986, ISBN 3-88745-075-2


```
 1:  ;-----------------------------------;
 2:  ;                                   ;
 3:  ;    Software-T-Switch für RS232    ;
 4:  ;    von Armin Krieglmeier          ;
 5:  ;        & GFA-Assembler 1.3        ;
 6:  ;    (c) MAXON Computer, 1990       ;
 7:  ;                                   ;
 8:  ;    XBRA-Kennung: COM2             ;
 9:  ;                                   ;
10:  ;-----------------------------------;
11:
12:  bios       = 13        ; TRAP's
13:  xbios      = 14
14:
15:  constat    = 1         ; BIOS-Funktionen
16:  conin      = 2
17:  conout     = 3
```

```
18:
19:  setexec    = 5
20:
21:  iorec      = 14
22:
23:  rs232      = 1         ; Devices
24:  midi       = 3
25:
26:  appl_init          = $0a000100    ;AES-Fkt-#n
27:  form_alert         = $34010101
28:  menu_register      = $23010101
29:  evnt_mesag         = $17000101
30:
31:             .TEXT
32:             lea.l     new_stack,sp    ;Stack
33:  init:                                 einrichten
```

→

```
 34:            move.l    #appl_init,d0 ;AES-Anmeldung
 35:            bsr       aes        ;ap_id j.in int_out
 36:            lea.l     addr_in,a6
 37:            lea.l     int_in,a5
 38:
 39:            move.w    int_out,(a5)
 40:            move.l    #my_name,(a6) ;in Menuleiste
 41:            move.l    #menu_register,d0
 42:            jsr       aes
 43:
 44:            move.w    int_out,my_id ;ac_id in my_id
 45:
 46:            pea       new_bios      ;neuen Disptchr
 47:            move.w    #45,-(sp)     ;installieren
 48:            move.w    #setexec,-(sp)
 49:            trap      #bios
 50:            addq.l    #8,sp
 51:            move.l    d0,old_vector ;alten sichern
 52:
 53:            move.w    #2,-(sp)      ;Iorec(2)
 54:            move.w    #iorec,-(sp)  ;liefert Adr.v.
 55:            trap      #xbios        ;MIDI-IOREC-Str
 56:            adda.l    #4,sp
 57:
 58:            movea.l   d0,a0         ;neue Pufferadr
 59:            move.l    #midi_buf,(a0) ;eintragen
 60:
 61:            move.l    #midi_end,d0  ;Pufferlänge
 62:            subi.l    #midi_buf,d0  ;berechnen
 63:            move.w    d0,4(a0)      ;und eintragen
 64:
 65: loop:
 66:            move.l    #evnt_mesag,d0
 67:            move.l    #msg_buf,(a6)
 68:            jsr       aes           ;Message warten
 69:
 70:            cmpi.w    #40,msg_buf   ;Accessory?
 71:            bne       loop          ;nein! Schade
 72:
 73:            move.w    msg_buf+8,d0
 74:            cmp.w     my_id,d0      ;meint er mich?
 75:            bne       loop
 76:
 77:            jsr       main       ;ja! Hauptroutine
 78:            bra       loop       ;und wieder warten
 79:
 80: main:      lea.l     alrt_txt(pc),a0 ;Adr AlertTxt
 81:            move.l    a0,(a6)    ;nach addr_in
 82:
 83:            move.w    device,d0
 84:            cmpi.w    #rs232,d0     ;dev=rs232?
 85:            bne       weiter
 86:            move.w    #2,(a5)       ;Ja, Midi deflt
 87:            bra       alert
 88: weiter:    move.w    #1,(a5)       ;Nein, RS232
 89:
 90: alert:     move.l    #form_alert,d0 ;Alertbox
 91:            bsr       aes
 92:
 93:            move.w    int_out,d0    ;Button nach d0
 94:
 95:            cmpi.w    #1,d0         ;RS232?
 96:            bne       midi?         ;nein, Midi
 97:            move.w    #rs232,device ;Dev ändern
 98: midi?:     cmpi.w    #2,d0         ;Midi nicht
 99:            bne       abbr?         ;->passiert nix
100:            move.w    #midi,device
101: abbr?:
102:            rts                     ;ins Desktop
103:
104:            .EVEN                   ;Anfang XBRA
105: xbra:      .DC.l 'XBRA'
106:            .DC.l 'COM2'
107: old_vector: .DC.l $ffffffff        ;alter
108:                                    ;Trap-Vektor
109:
110: device:    .DC.w rs232             ;Standardein-
111:                               ;stellung ist RS232
112: new_bios:
113:            movea.l   sp,a0
114:            btst      #5,(sp)    ;Supervisor-Mode?
115:            bne.s     label      ;- ja, dann weiter
116:            move.l    usp,a0     ;- nein, wir tun
117:            subq.l    #6,a0      ;so, als ob
118:
119: label:     cmpi.w    #conout,6(a0) ;wurde Conout,
120:            beq.s     change
121:            cmpi.w    #conin,6(a0)  ;Conin, oder
122:            beq       change
123:            cmpi.w    #constat,6(a0);Constat grufn?
124:            bne       call_bios     ;nein! Zum
                                        alten Dispatcher
125: change:    cmpi.w    #rs232,8(a0)  ;RS232?
126:            bne.s     call_bios     ;
127:                                    ;Wenn ja, dann
128:            move.w    device,8(a0)  ;Parms im Stack
129:                                    ;ändern.
130: call_bios:
131:            movea.l   old_vector,a1 ;ab ins BIOS
132:            jmp       (a1)
133:            rte
134:
135: ; Unterroutine für AES-Aufrufe
136: aes:
137:            movem.l   d1-d7/a0-a6,-(sp)  ;Register
138:            lea.l     contrl,a0          ;Control-
                                     Array initialisieren
139:            clr.l     (a0)
140:            clr.l     4(a0)              ; int_in-
                                         Array löschen
141:            movep.l   d0,1(a0)           ; neue
                                       Daten eintragen
142:            lea.l     aes_para_blk,a0
143:            move.l    a0,d1              ; Parame-
                                      terblock nach d1
144:            move.w    #$c8,d0            ; Kennung
                                               für AES
145:            trap      #2
146:            move.w    int_out,d0         ; Ergebnis
                                               nach d0
147:            movem.l   (sp)+,d1-d7/a0-a6  ; Register
                                          restaurieren
148:            rts
149:
150: ;-------------------------------
151:            .DATA
152: aes_para_blk:                           ; AES-
                                     Parameter-Block-
153:            .DC.l contrl                 ; Struktur
154:            .DC.l global
155:            .DC.l int_in
156:            .DC.l int_out
157:            .DC.l addr_in
158:            .DC.l addr_out
159:
160: my_name:   .DC.b "  RS232-Switch",0     ;AccEintr.
161:
162:                             ;Text für die Alertbox
163: alrt_txt:  .DC.b "[1]"
164:            .DC.b "[ RS232 Switch by
             A.Krieglmeier","| "
165:            .DC.b "|    Bitte wählen Sie Ihr      "
166:            .DC.b "|    Standard AUX-Device      ]"
167:            .DC.b "[RS232|MIDI|Abbruch]",0
168:
169: ;    Man könnte die Buttons natürlich auch
170: ;    COM1: und COM2: nennen.
171:
172: ;------------------------------------
173:            .BSS
174:
175: my_id:     .DS.w 1       ; Accessory_Id
176: msg_buf:   .DS.w 8       ; Nachrichten-Puffer
177:
178: global:    .DS.w 15      ; Aes-Datenfelder
179: contrl:    .DS.w 5
180: int_in:    .DS.w 16
181: int_out:   .DS.w 7
182:            .EVEN
183: addr_in:   .DS.l 2
184: addr_out:  .DS.l 65
185:
186: midi_buf:  .DS.l 1024    ; neuer Midi-Puffer (4 KB)
187: midi_end:
188:
189: new_stack: .DS.l $100   ; eigener Stack
190: ;------------------------------------
191:            .END
```

Listing 1: Das Treiberprogramm für die zweite serielle Schnittstelle

```
 1:  '
 2:  ' ##################################
 3:  ' ##          Listing 2           ##
 4:  ' ##   A.Krieglmeier / GFA-Basic  ##
 5:  ' ##################################
 6:  '
 7:  FUNCTION carrier
 8:    ' ----------------------------------------
 9:    '  liefert TRUE, wenn DCD auf COM2 high
10:    ' ----------------------------------------
11:    '
12:    LOCAL reg|,sel|,ret!
13:    '
14:    SDPOKE &H43E,1        !Disk-VBL deaktivieren
15:    SPOKE &HFF8800,14     !Soundchip Port A selekt
16:    reg|=PEEK(&HFF8800)   !alten Wert lesen
17:    sel|=BSET(reg|,7)     !DCD von COM2 selektieren
18:    SPOKE &HFF8802,sel|  !und ins Register schreiben
19:    '
20:    IF BTST(PEEK(&HFFFA01),1)   ! Bit 1 im MFP-GPIP
                                     gesetzt?
21:      ret!=FALSE                ! -> kein Carrier
22:    ELSE                        ! ansonsten
23:      ret!=TRUE                 ! Rück = false
24:    ENDIF
25:    '
26:    SPOKE &HFF8802,reg|         ! vorherigen DCD-
                                     Kanal auswählen
27:    SDPOKE &H43E,0              ! VBL wieder
                                     einschalten
28:    '
29:    RETURN ret!
30:  ENDFUNC
```

Listing 2: Dieses kleine Testprogramm testet den DCD

```
 1:  ' ##################################
 2:  ' ##          Listing 3           ##
 3:  ' ##   A.Krieglmeier / GFA-Basic  ##
 4:  ' ##################################
 5:  '
 6:  PROCEDURE dtr(flag!)
 7:    ' ----------------------------------------
 8:    '  setzt DTR auf log. high, wenn flag!=true
 9:    ' ----------------------------------------
10:    '
11:    LOCAL inh|
12:    '
13:    SDPOKE &H43E,1                ! Disk-VBL aus
14:    '
15:    SPOKE &HFF8800,14             ! Soundchip Port A
                                       selektieren
16:    inh|=PEEK(&HFF8800)           ! alten Wert lesen
17:    '
18:    IF flag!=TRUE                 ! DTR einschalten?
19:      inh|=BCLR(inh|,6)           ! IO_A6-Bit setzen
20:    ELSE                          ! oder auschalten
21:      inh|=BSET(inh|,6)           ! Bit löschen
22:    ENDIF
23:    SPOKE &HFF8802,inh|           ! neuen Wert ins Re-
                                       gister schreiben
24:    '
25:    SDPOKE &H43E,0                ! ... und VBL
     wieder aktivieren
26:  RETURN
27:  '
```

```
 1:  ' ########################################
 2:  ' ##    Test-Programm für Listing 1 & 2  ##
 3:  ' ##       DTR_PROC.LST & DCD_FUNC.LST   ##
 4:  ' ##       dazuMERGEn                    ##
 5:  ' ########################################
 6:  '
 7:  REPEAT
 8:    '
 9:    IF @carrier=TRUE                        !---------
10:      PRINT AT(10,10);" CARRIER on "        !
11:    ELSE                                    !  testet
                                                COM2-DCD
12:      PRINT AT(10,10);" CARRIER off"        !
13:    ENDIF                                   !---------
14:    '
15:    dtr(TRUE)                !-------------------------
16:    PAUSE 25                 !
17:    '                        !     läßt DTR-LED blinken
18:    dtr(FALSE)               !
19:    PAUSE 25                 !-------------------------
20:    '
21:  UNTIL INP?(2)
```

Listing 3 (links): Dieses kleine Testprogramm testet den DTR

Listing 4 (oben): Diese Testroutine überprüft, ob alles richtig ist



+3C

jetzt auch als Co-Prozessor Version verfügbar

Ein paar Worte zum Wahnsinn:

Um es gleich deutlich zu machen: Es geht hier um das Programm **SKYPLOT**, genauer um die neueste Version **SKYPLOT PLUS 3**.

Es gab einmal einen „absoluten Wahnsinn“ in Form von **SKYPLOT PLUS**, der dann dem „gesteigerten Wahnsinn“ in Gestalt des Nachfolgers **SKYPLOT PLUS 2** weichen mußte. Nun fällt uns leider keine weitere Steigerung mehr zum Wahnsinn ein, wo **SKYPLOT PLUS 3** diese doch verdient hätte! Was tun?

Wir lassen also die kühlen Fakten für sich sprechen, ganz vernünftig:

„...the most sophisticated astronomical simulation package that we have ever seen on a microcomputer“
(_The_Planetarian_)

„...die Möglichkeiten sind selbst bei häufiger Benutzung kaum auszuschöpfen“
(c't)

„...gäbe es einen Oscar oder eine goldene Palme für Atari-Programme, wäre SKYPLOT ein Kandidat dafür... ein echter Grund, sich einen Atari ST zuzulegen“
(XEST)

„...ein unglaubliches Programm... einfacher und eindrucksvoller läßt sich einem Interessierten der Kosmos kaum näherbringen“
(ATARI Magazin)

„...goldenes Byte für SKYPLOT“
(Computer persönlich)

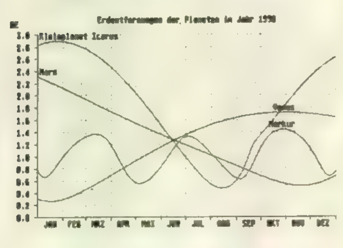
„...von dem Programm sehr begeistert... erwies sich auch im professionellen Gebrauch sehr nützlich“
(Max-Planck-Institut für Radioastronomie Bonn)

„...if you own an Atari and enjoy astronomy, you must get this program“
(Sky & Telescope)

Nun genug der Meinungen, schließlich beziehen sich die Pressestimmen ja alle auf die *alten* Versionen, denen zu **SKYPLOT PLUS 3** mindestens über ein Jahr an Entwicklungszeit fehlen!

Irgendwo muß diese Arbeit wohl stecken, und zwar hier:

- bis zu 64000 Sterne
- bis zu 32000 Nebel etc.
- umfangreiche On-Line-Hilfen
- Echtzeitsimulation
- jede Mange Diagramme und Simulationen zur Verdeutlichung von astronomischen Sachverhalten
- Bilder laden oder speichern: IMG, AIM, STAD, Neochrome, Degas (Elite), komprimiert



- eingebaute DOS-Funktoinen: Dateien löschen, Programme starten, Formatieren
- umfangreicher Parameter für Grafik und Drucker
- Zeichnen, Speichern und Laden von Bewegungsbahnen
- Plotausgabe, auch für HPGL-kompatible Plotter
- Laden von Daten stark beschleunigt

- Finsterniskanon
- Benutzung der Hardwareuhr
- Zonenzeit und Zeitzonen
- direkte Steuerung von Vergrößerung, Ausschnitt etc. durch Tasten
- 3D-Darstellung, auch für Stereoskope
- Grafiken bearbeiten: Spiegeln, Beschriften etc.
- Speichern von Sequenzen





- eingebaute Editoren für Textdateien und Sternbildhilfslinien
- Datenausgabe auf Drucker: Ephemeriden, Kalender etc.
- jede Menge Hardcopies eingebaut, auch für 24- oder 48-Nadeldrucker
- mitgelieferter Bitmaptreiber erreicht Auflösungen von 6912★4320 Pixeln oder mehr

Schluß mit dem Wahnsinn — her mit SKYPLOT PLUS 3C !

* alle Preise sind unverbindlich empfohlene Verkaufspreise

BESTELL-COUPON

Bitte senden Sie mir

___St. Skyplot plus 3	DM 198.–
___St. Skyplot plus 3 (Co-Prozessor Version)	DM 248.–
___St. Up-Date auf Skyplot plus 3 ohne Modul	DM 50.–
___St. Up-Date (von Skyplot plus 3 auf Co-Prozessor Version) mit Modul	DM 80.–
___St. Up-Date (von Skyplot plus 3 auf Co-Prozessor Version) ohne Modul	DM 50.–
___St. Up-Date von älteren Versionen jeweils Mehrpreis von	DM 50.–
___St. Up-Date auf Skyplot plus 3 mit Modul	DM 80.–

zuzügl. Versandkosten DM 6,– (Ausland DM 10,– unabhängig von der bestellten Stückzahl

Name, Vorname ____________

Straße, Hausnr. ____________

PLZ, Ort ____________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

Schweiz
Data Trade AG
Landstr. 1
CH - 5415 Rieden - Baden



Seit Veröffentlichung der Host-Adapter-Serie sind ja schon etliche Wochen ins Land gezogen und es wird Zeit, über die bisherigen Erfahrungen und über neue Ideen zu berichten. Die Reaktionen auf diese Schaltung waren bisher äußerst positiv, wie ich den Gesprächen mit Ihnen auf der CeBIT oder einigen Leserbriefen entnehmen konnte. Ich will aber nicht verschweigen, daß inzwischen auch 1 (in Worten: ein) Problem entdeckt wurde.

Das Problem

Beim Betrieb einiger Festplatten der Marke Quantum traten Schreibfehler auf; das Lesen funktionierte weiterhin fehlerfrei. Damit stand fest, daß die eigentliche Ursache im DMA-Chip des ATARI-Rechners zu suchen sei.

Wenn Sie sich das aktuelle Listing für das GAL IC5 im März-Heft (Seite 148) noch einmal ansehen, so finden Sie in Zeile 9 die Bemerkung 'FAST-MODUS'. In diesem Sinne schaufelt der Hostadapter die Daten schnellstmöglich über die Platine. Der DMA-Chip verträgt dies aber nur, wenn der Platten-Controller zwischen der Übertragung eines Bytes und der Anforderung des nächsten einige Dutzend Nanosekunden Pause einlegt. Genau das haben die damals getesteten Platten (vornehmlich SEAGATE) auch alle getan. Irgendwann tauchte dann die erste Quantum-Platte auf - und vorbei war's mit dem Fast-Modus.

Die Lösung

Halt! Sie denken jetzt bestimmt, die Tage des schnellsten, kompatiblen Host-Adapters seien damit gezählt. Mitnichten. Erstens treten diese Probleme nur beim Schreiben auf, das Lesen wird also gar nicht beeinflußt (GAL sei Dank). Zweitens erreicht meine persönliche Festplatte exakt die gleiche Schreibgeschwindigkeit wie im Fast-Modus - da gleicht der Platten-Controller aus, was ihm an Wartezeit aufgezwungen wird. Und drittens bleibt Ihnen ja bei den meisten Geräten die freie Wahl zwischen FAST- und neuem NORMAL-MODUS (einfallsreicher Name, nicht?).

Eigentlich ist diese Normalversion gar nicht neu, der erste Prototyp arbeitete noch damit. In der Annahme, sie wäre unnötig, gab es aber keine entsprechende Veröffentlichung. Sorry. An alle Quantum-Besitzer: Sie können die Fehlersuche beenden, die Lösung folgt sofort. Übrigens sind diese Platten, trotz des extremen Handshakings, auch nicht schneller als vergleichbare andere.

Normal-Modus V2.5 (Änderung für IC5 fett):

```
SACK = (REQ + RST + /SELFF & /A1 & /
COMP & /MSG)
& ACKFF & (/ACK + IO);
```

Hardware-Lock

Die Gespräche mit Lesern zeigten, daß der Schalter für den 'extended Mode' keinen großen Anklang fand. Meist wurden die entsprechenden Pins überbrückt und der Schalter einfach weggelassen. Zu Recht, wie ich meine; den Sinn dieser Umschaltung hatte ich ja selbst schon angezweifelt.

Doch wer sagt denn, daß diese Schaltfunktion nicht auch eine andere Bedeutung erlangen kann? Wie wäre es, wenn Sie Ihre Geräte einfach abschließen und so vor unberechtigtem Gebrauch schützen würden? Oder Sie könnten einen zuschaltbaren Hardware-Schreibschutz realisieren, durch den Ihre Daten auch dann erhalten bleiben, wenn der Nachwuchs im SED auf den 'Format'-Button drückt? Selbst der hartnäckigste Virus und die gesetzloseste Raubkopie (nana!) wären damit chancenlos.

Genau diese Möglichkeiten stehen uns aber (fast) problemlos zur Verfügung. Damit Sie die Änderungen in den GAL-Programmen nicht immer widerstandslos hinnehmen müssen, werden die möglichen Variationen einmal genauer als bisher beschrieben.

Die folgenden Vorschläge beziehen sich diesmal auf das IC2 und nicht, wie bisher, auf das IC5. Letzteres sollte aber eine aktuelle Version besitzen, also V2.4 (FAST) aus der ST-Computer 3/90 oder V2.5 (SLOW) wie oben beschrieben.

Als erstes wird die bisherige Schalterprogrammierung rückgängig gemacht. In allen Formeln, in denen TS1 vorkommt, entfällt diese Variable. Also: In SID1, SID2, STFF, DB5, DB6 und DB7 bitte '& TS1' bzw. 'TS1 &' ersatzlos streichen. Damit arbeitet der Host-Adapter nur noch im 'extended Mode', und der Eingang mit der Schaltfunktion TS1 steht zu unserer freien Verfügung.

Schließ' ab die Platte

Die einfachste Variante besteht darin, die Platte(n) per Schalter (Schlüsselschalter wäre ideal) zu blockieren. Das zentrale Signal auf dem Adapter heißt SELFF. Wird dieses Flipflop nicht aktiviert, erfolgt auch keine Selektion der angeschlossenen Geräte, d.h. für den Rechner sind gar keine Geräte vorhanden. Hier gilt es anzusetzen. Die Bedingung, die den Zugriff ermöglicht, wird also erweitert:

"Nur, wenn der (Schlüssel-)Schalter *nicht* betätigt ist, soll ein Zugriff stattfinden dürfen." Im Programm-Listing wird also die Formel für SELFF einfach um die Bedingung '& /TS1' erweitert.

Nun gilt dieser Schutz natürlich für alle Geräte, die an den Host-Adapter angeschlossen sind. Wollen Sie nur ein oder

zwei der vier möglichen Controller sichern, werden die Signale SID1 und SID2 verwendet, die die momentan angewählte SCSI-ID nach folgender Tabelle beinhalten:

SID2	SID1	SCSI-ID
0	0	0
0	1	1
1	0	2
1	1	3

Wieder muß in der Bedingung formuliert werden, wann ein Zugriff möglich sein soll und nicht wie er verhindert wird. Dazu ein Beispiel: Nach Betätigen des Schalters sollen die Wechselplatte (SCSI-ID 2) und der Streamer (SCSI-ID 3) für den Rechner nicht mehr vorhanden sein. Die Bedingung lautet diesmal: "Der Zugriff ist erlaubt, wenn entweder TS1 inaktiv ist oder wenn das Flipflop SID2 nicht gesetzt wurde, also die Geräte 0 oder 1 angesprochen sind." Die Formel für SELFF wird also diesmal um '& (/TS1 + /SID2)' erweitert.

Wer sich an diese Denkweise nicht gewöhnen kann, freundet sich wohl eher mit folgendem Vorgehen an: "Der Host-Adapter soll *nicht* reagieren, wenn der Schalter betätigt und das Signal SID2 gesetzt ist." Bezeichnen wir die bisherige SELFF-Bedingung mit B, lautet die Formel:

```
/SELFF = /B + TS1 & SID2;
```

Durch Invertierung beider Seiten folgt daraus:

```
    SELFF = B & /(TS1 & SID2);
und SELFF = B & (/TS1 + /SID2);
```

Es ist also völlig egal, auf welche Art und Weise eine Bedingung formuliert wird. Hauptsache, das gewünschte Ergebnis läßt sich erreichen.

And now for something completely different

Wird die mit 'B' bezeichnete Bedingung einmal manuell oder mit Hilfe eines GAL-Assemblers ausmultipliziert, ergeben sich vier 'oder-'verknüpfte Terme. Mit Verwendung von '& (/TS1 + /SID2)' aus dem letzen Beispiel erreichen wir bereits die maximal mögliche Anzahl von acht Termen bei einem 'registered' GAL-Ausgang. Für einen vernünftigen Schreibschutz sind aber weitere Bedingungen erforderlich, die mit der jetzigen Struktur nicht realisierbar wären.

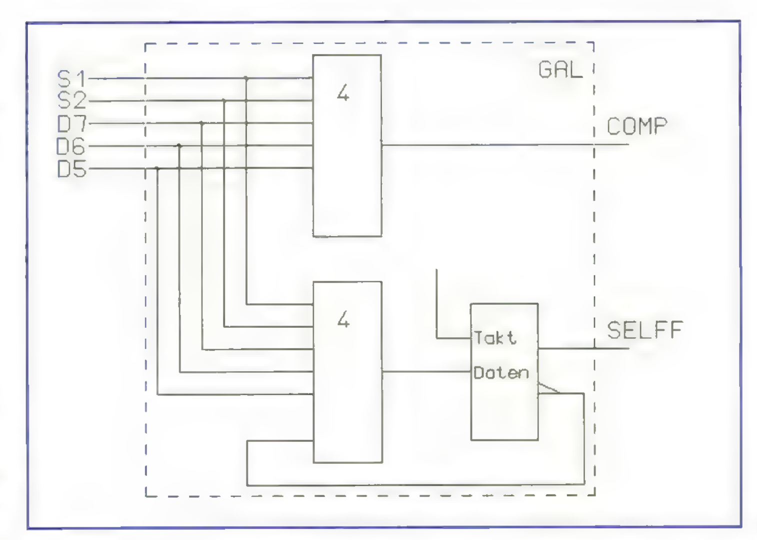


Bild 1a

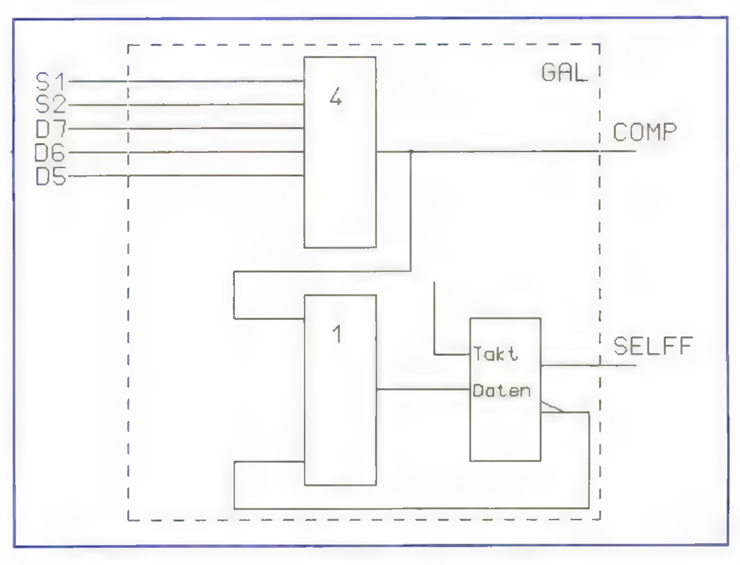


Bild 1b

Der Konjunktiv deutet es bereits an - das Problem ist lösbar. Das Signal COMP besitzt eine ähnliche Struktur wie unser SELFF, und es spricht nichts dagegen, diese Verknüpfung ebenfalls zu verwenden. Bild 1a deutet den Signalverlauf im GAL vor und Bild 1b nach der Umstrukturierung an (keine Pickel mehr). Die Kästen enthalten kombinatorische Logik aus Und-/Oder-Gattern. Außerdem ist die Anzahl der verwendeten Terme angegeben. Die neue 'nackte' SELFF-Formel lautet jetzt " SELFF = COMP & /SELFF ", besteht demnach nur noch aus einem einzigen 'Und'-Gatter.

Ich höre Sie schon murmeln: "Warum hat der Blödmann das nicht gleich so gemacht?". Nun, das Vergleichssignal für die Target-Nummer legt jetzt einen wesentlich größeren Weg zurück bis es das Flipflop SELFF erreicht. Damit wird die Setup-Zeit - das ist die Differenz zwischen Datensignal und der folgenden Taktflanke - auf etwa die Hälfte reduziert. Das GAL wird zwar weiterhin innerhalb seiner Spezifikation betrieben, doch halte ich es für einen schlechten Entwicklungsstil, grundlos bis an die Grenzen der Hardware zu gehen, und bisher gab es keinen Grund dafür. Wie dem auch sei, uns stehen jedenfalls sieben (!) freie Terme zur Verfügung.

Schreibschutz per Hardware

Das Prinzip dieses Schutzes ist ganz einfach. Soeben wurden Geräte definiert, auf die (nicht) zugegriffen werden kann. Jetzt werden Befehle definiert, die den Geräten (nicht) gesendet werden dürfen. Befehle bestehen bei diesem Host-Adapter aus den fünf Kommando-Bits D0 bis D4 und aus der Kommandogruppe DB5 bis DB7. Letztere dürfen nicht mit D5 bis D7 verwechselt werden, denn das ist die Target-Nummer.

Soll z.B. ausschließlich der Befehl MODE SENSE ($1A) zugelassen werden, lautet die entsprechende Formel:

```
SELFF = COMP & /SELFF & ( /TS1 +
        /DB7 & /DB6 & /DB5 & D4 & D3 &
        /D2 & D1 & /D0);
```

Der prinzipielle Aufbau entspricht genau den bisher beschriebenen Bedingungen, und die Daten-Bits ergeben binär %00011010, d.h. hexadezimal $1A. Null Problemo, oder?

Eigentlich soll die Festplatte auch noch ein paar andere Befehle verstehen, nicht nur MODE SENSE. Genaugenommen soll sie mit Ausnahme der Schreibbefehle, wie z.B. FORMAT, WRITE oder MODE SELECT, ja ganz normal arbeiten. Bei meiner SEAGATE-Platte sind es immerhin 20 Befehle, die weiterhin verfügbar sein sollten. Zwanzig Terme sind jedoch ziemlich genau dreizehn Terme zuviel. Was tun?

```
******************************************
 Dokumentation: BOOLEAN-EQUATIONS
 Datei: SC02V25.PLD
 Datum: 04.04.90     Uhrzeit: 14:27
******************************************

SCSI-INTERFACE
IC2,    GAL FÜR DATENWEGE
'Extended Mode' ständig aktiv
Hardware-SCHREIBSCHUTZ für alle Geräte
by Reiner Wiechert
Copyright MAXON Computer 1990

*IDENTIFICATION
 SC02V25;

*TYPE
 GAL20V8;

*PINS
%CSA1   =  1,          Takt                    %
 /S1    =  2,        % Target-Nummer MSB       %
 /S2    =  3,        % Target-Nummer LSB       %
 D7     =  4,        % Datenbit 7              %
 D6     =  5,        % Datenbit 6              %
 D5     =  6,        % Datenbit 5              %
 D4     =  7,        % Datenbit 4              %
 D3     =  8,        % Datenbit 3              %
 D2     =  9,        % Datenbit 2              %
 D1     = 10,        % Datenbit 1              %
 D0     = 11,        % Datenbit 0              %
 /TS1   = 14,        % Umschalter              %
 /CSA1  = 23,        % Takt als Dateneingang %
  /STFF.R  = 15,      % Merker: 'Zusatzinformation vorhanden'     %
 /SELFF.R = 16,      % Merker: 'Hostadapter selektiert'           %
 SID1.R   = 17,      % SCSI-ID LSB                                %
 SID2.R   = 18,      % SCSI-ID MSB                                %
 DB5.T    = 19,      % SCSI-Datenbit 5                            %
 DB6.T    = 20,      % SCSI-Datenbit 6                            %
 DB7.T    = 21,      % SCSI-Datenbit 7                            %
 COMP.T   = 22;      % Vergleich Daten <-> Target-Nummer erfüllt %

*BOOLEAN-EQUATIONS

  SID1   = (S2 & D7 + /S2 & /D7) & (S1 & D6 + /S1 & /D6) & /SELFF
         & (D5 & D3 + /D5 & SID1 & STFF);

  SID2   = (S2 & D7 + /S2 & /D7) & (S1 & D6 + /S1 & /D6) & /SELFF
         & (D5 & D4 + /D5 & SID2 & STFF);

  STFF   = (S2 & D7 + /S2 & /D7) & (S1 & D6 + /S1 & /D6) & /SELFF
         & D5;
  DB5    = SELFF & /CSA1 & D5  +  CSA1 & DB5 +  (STFF & DB5
         + /SELFF & /STFF & D0 &
           (S2 & D7 + /S2 & /D7) & (S1 & D6 + /S1 & /D6) & D5);

  DB6    = SELFF & /CSA1 & D6  +  CSA1 & DB6 +  (STFF & DB6
         + /SELFF & /STFF & D1 &
           (S2 & D7 + /S2 & /D7) & (S1 & D6 + /S1 & /D6) & D5);

  DB7    = SELFF & /CSA1 & D7  +  CSA1 & DB7 +  (STFF & DB7
         + /SELFF & /STFF & D2 &
           (S2 & D7 + /S2 & /D7) & (S1 & D6 + /S1 & /D6) & D5);

  SELFF  = COMP & /SELFF & (/TS1 +  /DB7 & /DB6 & /DB5 & D4 & D1
         + DB7 & DB6 & DB5 & /D4 & /D3 & D2 & /D1 & D0
         + /DB7 & /DB6 & /D2 & /D1  +  /DB7 & /DB6 & DB5 & D2
         + /DB7 & /DB6 & /DB5 & D4 & D3 & D2 & /D1
         + /DB7 & /DB6 & /D4 & /D2 & D1 & D0);

  COMP   = (S2 & D7 + /S2 & /D7) & (S1 & D6 + /S1 & /D6) & /D5;

*END
```

Quine-McCluskey, hilf!

Wer bei acht Eingangsvariablen, nämlich den Daten-Bits, noch eine Karnaugh-Tafel auswerten kann, möge sich bitte melden. Der Nobelpreis wartet. Normalsterbliche müssen sich allerdings nach anderen Möglichkeiten umsehen.

In dieser, unserer ST-Computer erschien kürzlich eine Artikelreihe, die ein Lösungsverfahren für das Problem beschrieb [1]. Also schnell ein Programm geschrieben und in freudiger Erwartung dem Ergebnis entgegengefiebert. (Der dritte Teil mit dem Programmbeispiel war noch nicht veröffentlicht.) Das QMC-Verfahren schaffte es tatsächlich, die Anzahl der Terme auf 11 zu reduzieren. Das war aber nicht genug.

Zwei Möglichkeiten boten sich an: Entweder mußte die Anzahl der zugelassenen Befehle reduziert oder erweitert werden. Beides *kann* zu einer Vereinfachung führen. Da ich auf keinen der liebgewonnenen Befehle verzichten wollte, blieb nur die Erweiterung. Das bedeutet natürlich nicht, daß beispielsweise der Formatierbefehl wieder zugelassen werden sollte, sondern daß auch nicht benutzte Kommandos, wie zum Beispiel $02, in die Vereinfachung eingehen. Das ist auch nicht weiter tragisch, denn der Platten-Controller würde einen solchen Befehl mit der hämischen Bemerkung 'illegal request' abweisen.

Leider kann niemand sagen, welche Opcodes zu einer weiteren Vereinfachung führen und welche die Situation nur verschlimmern. Das Verfahren, mit dem es dann doch ging, funktioniert so: Für alle Opcodes, die keinem Schreibbefehl entsprechen, erhält die Ausgangsvariable den Zustand 1. In diesem Fall besteht die disjunktive Normalform aus 249 (!) Minitermen, da mein Controller nur sieben schreibende Befehle kennt (8 Bit -> 256 Zeilen - 7 = 249).

Die folgende Optimierung ist ein Härtetest für jedes QMC-Programm. Von den existierenden 249 einschlägigen Indizes benötigen wir nur noch die 20, die den gewünschten Befehlen entsprechen. Der Rest wird vergessen. Aus den Primimplikanten sucht man sich lediglich diejenigen heraus, die zur Erfüllung der Vorgaben benötigt werden. Das Ergebnis überzeugt: Nur noch fünf Terme sind notwendig, um alle Vorgaben zu erfüllen.

Dieses Verfahren funktioniert natürlich nur mit Programmen, die als Ergebnis eine Tabelle aus Primimplikanten und einschlägigen Indizes erstellt (siehe [1]). Erzeugt das Programm 'nur' die fertige Lösungsformel, ist es für diese Anwendung unbrauchbar. Bisher fehlt noch geeignete Software, die als Ausgangsvariable außer Nullen und Einsen auch ein 'X' für 'beliebig' zuläßt.

Ich habe das Resultat noch etwas überarbeitet, um die Anzahl der unbenutzten Opcodes noch zu reduzieren. Die Formel für SELFF, die den üblichen SCSI-Kommandosatz abdeckt, finden Sie in Listing 1 abgedruckt. Betätigen Sie bei dieser Programmierung den Schalter TS1, wird jeder Schreibversuch mit einem TIMEOUT quittiert. Wundern Sie sich also nicht über Meldungen wie "Daten eventuell defekt" oder "Create error". Wordplus meldet sogar korrekt "Diskette schreibgeschützt". Auf diese Weise läßt sich gut ermitteln, wann ein Programm schreibend auf ein Medium zugreift (s. Listing).

Nicht zu vergessen, daß auch eine Verknüpfung aller bisher beschriebenen Schutzarten möglich ist. Schreibschutz nur für ganz bestimmte Geräte, wechselweiser Schutz je nach Schalterstellung oder ähnliche Mechanismen sind realisierbar, solange die magische 'Acht-Terme-Grenze' nicht überschritten wird. Außerdem stehen nur zwei Schaltstellungen zur Verfügung. Heißer Tip am Rande: Wenn Sie sich per GAL-Programm auf ein oder zwei Target-Nummern festlegen, können mit den Schaltern S1 und/oder S2 weitere Kombinationen verwirklicht werden. Ich bin gerne bereit, Sie bei der Realisierung neuer Ideen zu unterstützen.

Sonstiges

Es gibt einige Fragen, die häufiger gestellt werden und ich nutze die Gelegenheit, diese möglichst allgemein zu beantworten.

Manche Selbstbauer hatten Probleme, den Host-Adapter zum Leben zu erwekken und hofften auf Druckfehler in der Veröffentlichung. Der Schaltplan und das Layout sind jedoch direkt den Originalunterlagen entnommen und daher fehlerfrei wiedergegeben. Fehler können bei derartigen Schaltungen an vielen Stellen auftreten, doch haben viele Leser bereits bewiesen, daß sich aus den Vorlagen ein einwandfrei funktionierendes Gerät aufbauen läßt.

Die Frage nach dem 'Dual-Modus' eines Konkurrenzproduktes wurde auf der CeBIT gestellt. Manche Leser verwirrte wohl die Werbeaussage: "Ein gepuffertes DMA-Gerät arbeitet nicht, wenn sein IN-Anschluß an ein Parallelgerät angeschlossen wird". Parallel bedeutet hier ungepuffert. Wie im zweiten Teil dieser Artikelreihe bereits ausführlich beschrieben wurde, arbeitet unser MAXON-Host-Adapter selbstverständlich auch in dieser Betriebsart.

Zum Abschluß noch einen wichtigen Hinweis: Die veröffentlichten GAL-Programme beziehen sich auf die ebenfalls veröffentlichte Hardware V2.2. MAXON verwendet ein an den Mega ST angepaßtes Layout mit leicht geänderter Beschaltung (V2.3). Bestücken Sie also bitte keine gekauften Platinen mit selbstprogrammierten GALs oder umgekehrt. Es funktioniert nicht. Ansonsten wünsche ich weitere, frustlose Bastelstunden.

Reiner Wiechert

[1] *ST-Computer 2/90 und 3/90: Logische Schaltungen; Rainer Storn*

Kometen und Relativität

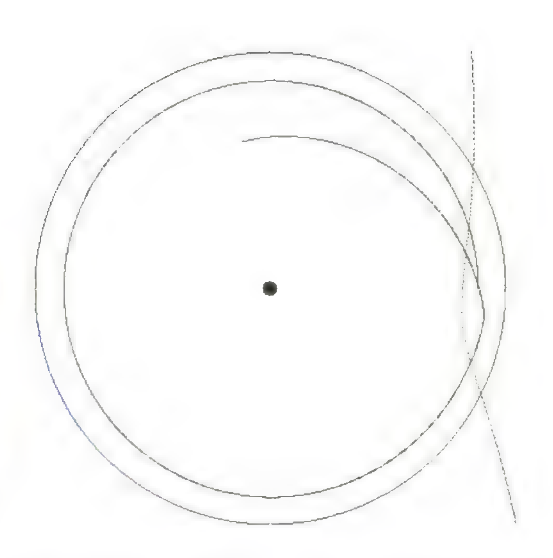


Bild 1: Ein (schwerer) Komet, der einem Sonnensystem zu nahe kommt, kann die umlaufenden Planeten in ihrer Bahn stören, ohne selbst wesentlich abgelenkt zu werden.

Das Weltall birgt viele Geheimnisse, die zum Teil sehr interessant, aber auch hochkomplex sein können. In diesem Artikel möchte ich Ihnen zwei Effekte vorstellen, deren mathematische Beschreibung zunächst komliziert erscheint, die sich mit einem Rechner aber leicht simulieren lassen. Bevor ich auf diese Effekte eingehe, möchte ich Ihnen zuerst einen Einblick in die Beschreibung der Gravitationskraft geben.

Die Gravitationskraft ist die Kraft, die unser Sonnensystem, die Galaxien und das gesamte Weltall zusammenhält. Sie wirkt auf alle Körper, wie z.B. Planeten, Galaxien, aber auch Äpfel oder (leider) auch Blumenvasen. Die Stärke der Kraft zwischen zwei Körpern läßt sich einfach über das Gravitationsgesetz berechnen:

```
Kraft= Konstante*M(K1)*M(K2)/
Quadrat[Abstand(K1,K2)]
```

Hierbei stehen "K1", "K2" für die beiden Körper, und "M(K)" soll die Masse ("Gewicht") des jeweiligen Körpers symbolisieren. Die Konstante ist die allgemeine Gravitationskonstante (G), die sich leicht über Fallversuche bestimmen läßt. Da Massen immer positiv sind, ist die Gravitationskraft immer anziehend. Dies im Gegensatz zur elektrischen Kraft, bei der sowohl positive als auch negative Ladungen und damit auch anstoßende Wirkungen existieren.

Möchte man aber nun wissen, wie mehrere Körper auf unser Objekt wirken, muß man die einzelnen Kräfte einfach addieren. Hierbei ist jedoch darauf zu achten, daß man die Richtung berücksichtigt, in der die Kräfte wirken. Das wird möglich, wenn man von "oben" auf die Geschehnisse schaut, die in einer Ebene ablaufen. Ähnlich den Pixeln auf dem Bildschirm läßt sich nun jedem Punkt ein Wert auf der x- und y-Koordinate zuordnen. Dieses Koordinatensystem (vgl. Zeichnung) wird im Programm im Datentyp "vector" durch die Elemente "x" und "y" dargestellt.

Für die Kraft ergibt sich:

```
Kraft_X=G*M(K1)*M(K2)*delta_X/
[Abstand(K1,K2)]^3
```

Hier ist nun:

```
delta_X=x(K1)-x(K2)
die Differenz der x-Koordinaten
Abstand(K1,
K2)=Wurzel[Quadrat(delta_x)+Quadrat(delta_y)]

(...)^3=(...)*(...)*(...)
```

Für die y-Koordinate gilt entsprechendes. Im Programm werden diese Berechnungen in der Funktion "get_besch" ausgeführt. Dort wird die Masse des Körpers nicht berücksichtigt, da sie sonst in "make_aenderung" wieder wegdividiert werden müßte. Die Werte von Kraft_X bzw. Kraft_Y können nun negativ und positiv werden, da sie auch Richtungsinformation beinhalten. Ein negativer Wert bedeutet, daß er gegen die Achsenrichtung zeigt. Das Verständnis für diese mathematische Darstellung ist nicht einfach, sie wird anschaulicher, wenn Sie ein wenig mit dem Programm experimentieren.

Ihre Wirkung

Die Kraft ist uns also bekannt, aber wie erhält man die Bewegung? Nun, wenn man auf einen Körper eine Kraft wirken läßt, z.B. dadurch, daß man ein Auto anschiebt, ändert sich seine Geschwindigkeit:

```
Änderung(Geschwindigkeit_X)=Kraft_X/
M(K)*Wirkzeit
```

Das M(K) wird in "make_aenderung" weggelassen (vgl. oben). Diese Änderung muß zur vorhandenen Geschwindigkeit hinzuaddiert werden. Mit der Geschwindigkeit ändert sich nun auch der Ort:

```
Änderung
(Ort_X)=Geschwindigkeit_X*Wirkzeit
```

Auch diese Änderung muß aufaddiert werden, was in "make_aenderung" durchgeführt wird. Die Wirkzeit ist hierbei die Dauer, über die die Kraft wirkt. Diese darf sich innerhalb dieses Zeitraumes aber nicht ändern. Sobald sich die Positionen der Körper auch nur ein bißchen verschieben - also sofort -, ändert sich auch die Kräfteverteilung zwischen den Körpern. Die Wirkzeit wäre demnach unendlich kurz und man müßte unendlich oft und damit unendlich schnell mit den neuen Positionen die neuen Kräfte berechnen, wieder neue Positionen, neue Kräfte usw.

Zum Glück reicht es aber, wenn dieser Zeitraum endlich gewählt wird. Innerhalb gewisser Grenzen sind die Fehler dann so gering, daß sie keine negativen Folgen haben. Die Größe dieses Zeitraums hängt hierbei weitgehend vom Problem ab und

kann über viele Größenordnungen von Sekunden bis Jahre reichen (im Programm wird die Zeit immer in bezug auf 86400 Sekunden angegeben, was einem Tag entspricht). Vor allem wenn sich zwei Objekte sehr nahe kommen und sie sich daher recht schnell bewegen, sind kleine Wirkzeiten notwendig. Entfernen sie sich dann wieder voneinander, könnte man eigentlich wieder ungenauer rechnen, um Rechnerzeit zu sparen. Im Programm bin ich aus zwei Gründen nicht darauf eingegangen. Erstens, damit das Programm gut verständlich bleibt, denn die notwendigen Wirkzeiten dürfen nicht ständig mit dem Abstand variieren, sondern können höchstens stufenweise verändert werden. Probieren Sie einmal, das zu implementieren. Zweitens kann man dann aus dem Abstand der Pixel und ihrer Laufgeschwindigkeit nicht mehr auf die Geschwindigkeit des Körpers schließen. Die benutzte Wirkzeit läßt sich dadurch überprüfen, daß man das Bild für verschiedene Zeiten berechnen läßt. Sind die Bilder (fast) gleich, können sie als richtig angenommen werden.

Der Komet kommt

Nach diesen Vorbereitungen möchte ich Ihnen als erstes die Wirkung von schweren Kometen auf die Bahn von Planeten eines Sonnensystems präsentieren. Hierzu müssen Sie das abgedruckte Listing eintippen und es notfalls an Ihr Sprachsystem anpassen. Das Programm malt zunächst die Sonne in den Bildschirmmittelpunkt und dann einen Kreis in voller Bildschirmhöhe, der dem Betrachter zur besseren Orientierung dienen soll. Danach beginnt das Programm mit der Darstellung des Planeten. Zum Startzeitpunkt befindet sich dieser genau rechts von der Sonne, weshalb der Wert für Ort_Y in der Initialisierung 0.0 ist. Damit er nicht in die Sonne stürzt, ist eine gewisse Startgeschwindigkeit notwendig. Soll der Planet in einer Kreisbahn um die Sonne wandern, bestimmt sich seine Startgeschwindigkeit über:

```
Gesch=Wurzel[G*M(Sonne)/
Startabstand(Planet,Sonne)]
```

In dem angegebenen Beispiel ergibt sich für diese Geschwindigkeit ein Wert von 55 [m/s]. Diesen Betrag Geschwindigkeit kann man nun auf die Koordinaten verteilen. Aber Vorsicht:

```
Betrag=
Wurzel(Quadrat(Gesch_x)+Quadrat(Gesch_y))
```

Im Beispiel zeigt diese voll in die negative y-Richtung. Besitzt der Planet nicht diese Geschwindigkeit, fliegt er auf einer Ellipsenbahn. Bei zu hohen Geschwindigkeiten kann es passieren, daß er das System verläßt, wie es beim Kometen der Fall ist.

Der Komet ist zu Beginn noch nicht zu sehen, da dieser aus dem Unendlichen (im Programm mit einer Entfernung von 760 Milliarden (e9) Metern, da wir nicht ewig warten können) auf das Planetensystem zufliegt. Was dann passiert, sehen Sie auf dem ersten Bild. Versuchen Sie doch selber, einige schöne Bahnen zu erhalten. Aber variieren Sie hierbei immer nur einen Wert, den Sie der Einfachheit halber im Programm abfragen lassen. Sie können auch ein vollständiges Planetensystem eingeben, wofür noch einige weitere Körper eingefügt und in der Hauptroutine die gegenseitigen Wirkungen berechnet werden müssen. Nach diesem Mehrkörperproblem noch ein kurzer Einblick in die Wirkung von relativistischen Effekten im Kosmos.

Merkur und Relativität

Im Sonnensystem bemerkten die Astronomen im 19. Jahrhundert einige Bahnstörungen (bei Merkur zirka ein Grad im Jahrtausend), die mit der Zeit durch die Entdeckung weiterer Planeten erklärt werden konnten. Die "Perihel-Drehung" des Merkur konnte somit (fast) erklärt werden: Die Ellipsenbahn des Merkur dreht sich um die Sonne. Wir haben also nicht nur die Bewegung des Merkur auf einer Ellipse um die Sonne, sondern auch noch die Drehung dieser Ellipse um die Sonne. Bei dem fehlenden Betrag der Drehung handelt es sich um einen Winkel von ungefähr einem Grad in 10000 (zehntausend) Jahren. Diese kleinen Beträge machen deutlich, warum die Störungen nicht schon eher bemerkt wurden. Als dann die "Allgemeine Relativitätstheorie" in ihren Grundzügen aufgestellt war, konnte man sich auch den noch fehlenden Betrag erklären. Bestimmt haben Sie schon etwas über Schwarze Löcher gehört. Diese haben eine so hohe Dichte, daß sie kleiner sind als der sogenannte Schwarzschildradius und daher alles Licht, das auf sie einstrahlt, festhalten und nicht wieder abgeben. Da sie alles aufsaugen und nichts wieder abgeben, wird ihr Name verständlich. Ein solcher Radius, der sich aus

```
Radius=G*M(K)/
Quadrat{Lichtgeschwindigkeit[3e8 (m/s)]}
```

errechnet, läßt sich auch der Sonne zuordnen. Er beträgt ca. 1.5km. Da sie aber viel größer als ihr Schwarzschildradius´ ist, kann sie dennoch leuchten, ist also kein Schwarzes Loch. Auch in ihrem Inneren befindet sich keines, da hierzu die Mittelpunktsdichte zu klein ist. Trotzdem ergibt sich daraus eine Änderung der Gravitationskraft, denn diese geht nun nicht mehr vom Mittelpunkt der Sonne aus, sondern vom Rand des Schwarzschildradius, der demnach von dem errechneten Körper-Körper-Abstand subtrahiert werden muß. Da dieser zu den tatsächlichen Abständen aber sehr klein ist, haben wir bisher fast keinen Fehler gemacht. Nun soll dieser Effekt aber berücksichtigt werden.

Um zu sehen, was passiert, müssen Sie nur noch die am Ende des Listings angegebenen Daten an die entsprechenden Stellen des Programms kopieren und dieses compilieren. Damit der Effekt sichtbar wird, wurde der Schwarzschildradius von

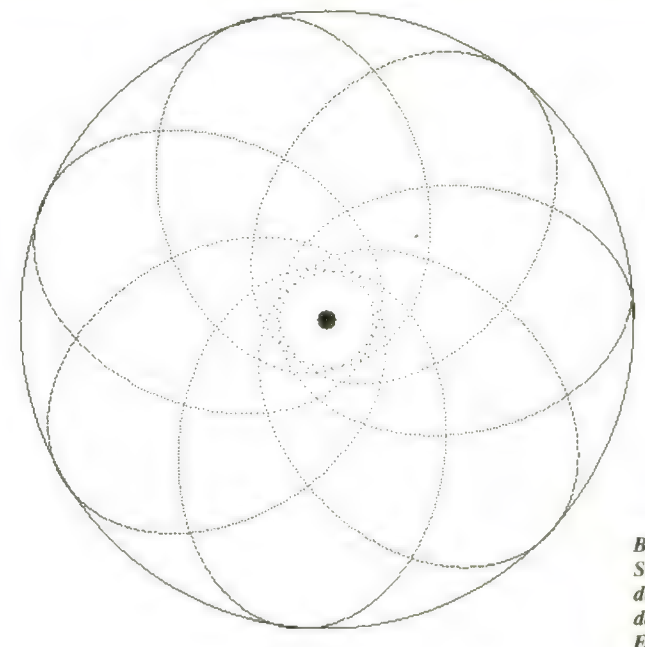


***Bild 2:** Durch eine einfache Subtraktion verändert sich die Gravitationskraft so, daß aus einer einfachen Ellipsenbahn eine Rosette wird.*

Hard&Soft

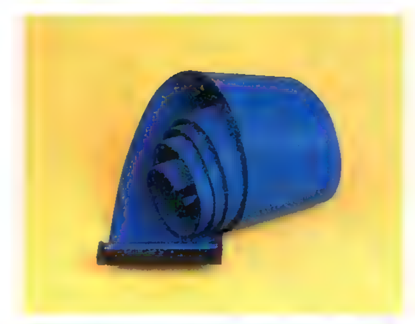
DAS VORBILD

Unser System setzt Maßstäbe

in Leistung, Zuverlässigkeit und Geschwindigkeit, ist kaum zu hören und natürlich vollkommen im Atari ST Design gehalten. Die ausgefeilte, grafische Benutzerführung macht die Bedienung spielend leicht.
... mit weniger Qualität sollten Sie sich nicht zufrieden geben.

Lieferbare Ausführungen

SCSI Ultra Speed Drive	
42 MB - 19 ms - 64 KB Cache	1498,–
85 MB - 19 ms - 64 KB Cache	1998,–
120 MB - 15 ms - 64 KB Cache	2498,–
170 MB - 15 ms - 64 KB Cache	2998,–
210 MB - 15 ms - 64 KB Cache	3498,–
SCSI Speed Drive	
49 MB - 28 ms	1398,–
85 MB - 24 ms	1798,–
Wechselplatte SCSI Speed Drive 44	1998,–
Streamer SCSI Speed Drive 155	2298,–

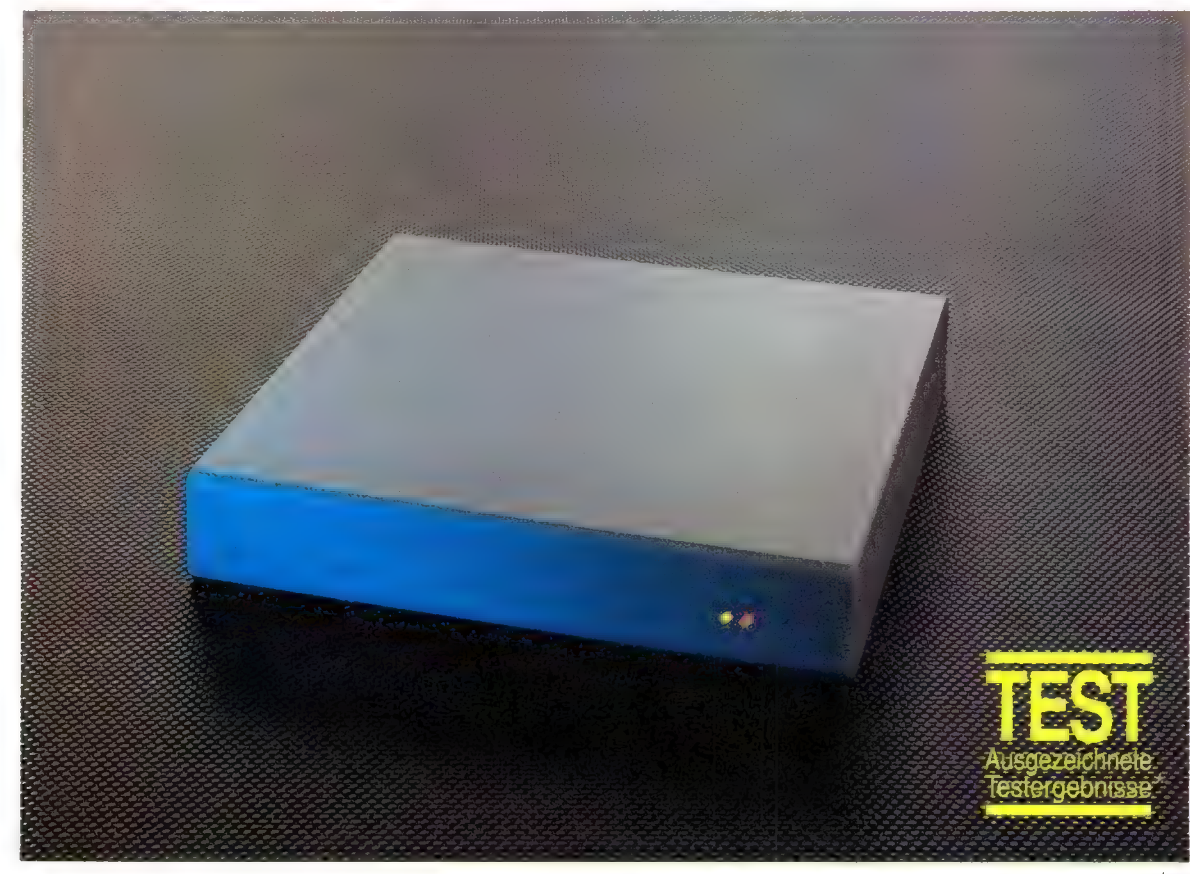


intern voller SCSI Standard



SCSI Schnittstelle an Geräterückseite herausgeführt

SCSI Ultra Speed Drive



* »... ein sehr gutes Preis/Leistungsverhältnis« (ST Computer 4/90)
»Fazit – ein Referenzmodell auf dem Festplattenmarkt« (ST Magazin 8/90)

Speichererweiterungen

Unsere Speichererweiterungen entsprechen dem neuesten Stand der Technik. Ingenieurmäßiges Schaltungsdesign, die Fertigung großer Stückzahlen auf hochmodernen Industriestraßen sowie ausgefeilte Maßnahmen zur Qualitätssicherung setzen einen hohen Qualitätsstandard, von dem auch Sie profitieren können.

Unsere Speichererweiterungen laufen in allen Rechnern. Durch gezielte Maßnahmen konnte die Stromaufnahme und die Störanfälligkeit bei knappem Bustiming deutlich reduziert werden. Dadurch können unsere Erweiterungen auch in Rechnern mit IMP-MMU (bei IMP-MMU nur gleich große Bänke möglich) oder in Atari 1040 ST problemlos betrieben werden.

Völlig neue Ideen bei Speichererweiterungen wurden bei unserer Multi-MB-Erweiterung gegangen. Die lange Erfahrung in Konstruktion und Entwicklung sowie das Bemühen um größtmögliche Flexibilität sind in die Entwicklung dieses Produktes eingegangen. Und das hat sich gelohnt.

Mit ein und derselben Karte können Sie zum Beispiel einen
– Atari 1040 ST von 1 MB auf 2 MB,
– einen Atari 260/520 von 512 KB auf 1 MB,
– einen Atari 260/520 von 512 KB oder 1 MB auf 2 MB,
– einen Atari Mega ST 1 von 1 MB auf 2 MB
– und
aufrusten, egal, ob in Ihrem Rechner eine IMP-MMU sitzt oder nicht.

NEUHEIT Multi-MB-Board

Sie wollen Ihren Rechner auf 4 MB erweitern? Kein Problem. Hierzu stecken Sie einfach eine Zusatzplatine Huckepack auf das Multi-MB-Board und schon haben Sie 4 MB. Einfacher und universeller geht es nicht.

Und dabei ist unsere Multi-MB-Erweiterung auch noch sparsam. Es werden immer nur soviele RAM's gesteckt, wie auch wirklich benötigt werden. Wurde früher eine voll mit 2 MB bestückte Platine benötigt, um einen 520 ST oder 1040 ST aufzurüsten, so benötigen Sie mit dem Multi-MB-Board nur 1 MB bzw. 1,5 MB um den Speicher aufzurüsten. Und wenn wir RAM's sparen – dann sparen Sie Geld.

Unser Angebot umfaßt eine große Palette an professionellen Speichererweiterungen, die generell in zwei Versionen lieferbar sind.

Die vollsteckbare Version ist kinderleicht einzubauen. Sämtliche Verbindungen der Speicherkarte können ohne Lötarbeiten vorgenommen werden. Voraussetzung für den Einbau dieser Version ist, daß im Rechner die MMU sowie der Videoshifter gesockelt sind. Der Steckverbinder zur MMU besitzt natürlich vergoldete Kontakte. Die teilsteckbare Version ist für Rechner mit nicht gesockelten Bauteilen (Shifter und/oder MMU) und für alle, denen das Anlöten von ca. 18 Lötverbindungen keine Probleme bereitet, gedacht.

Zum Lieferumfang jeder Speichererweiterung (bestückt) gehört eine ausführliche und bebilderte Einbauanleitung sowie ein Speichertestprogramm. Selbstverständlich wird jede unserer Speichererweiterungen vor dem Versand im Rechner stückgeprüft.

Sollte dennoch eine Frage offenbleiben, so helfen Ihnen an unserer Service-Hotline versierte Techniker gerne weiter.

Modell 1 S: Speicheraufrüstung auf 1 MB, voll steckbar, 198,00 DM

Modell 2: Speicheraufrüstung auf 2,5 MB, teilsteckbar (beim Mega ST 2 Aufrüstung auf 4 MB möglich), 449,00 DM; dto. Leerkarte 189,00 DM

Modell 2 S: wie Modell 2, aber voll steckbar, mit vergoldeten Mikrokontakten, 498,00 DM, dto. Leerkarte 229,00 DM.

Modell 2/4: Speicheraufrüstung auf 2,5/4 MB, teilsteckbar, Speicher in zwei Stufen 2,5/4 MB aufrüstbar. Auch für Mega ST geeignet, 498,00 DM; dto. als Leerkarte 249,00 DM.

Modell 2/4 S: wie Modell 2/4, aber voll steckbar, mit vergoldeten Mikrokontakten, 549,00 DM; dto als Leerkarte 298,00 DM.

Multi-MB-Board, teilsteckbar, bestückt zur Aufrüstung von 1 MB auf 2 MB, 398,00 DM; nicht als Leerkarte erhältlich!

Multi-MB-Board, vollsteckbar, bestückt zur Aufrüstung von 1 MB auf 2 MB, 449,00 DM, nicht als Leerkarte erhältlich!

Speichererweiterung für 1040 STE SIM-Module auf 2,5 MB 449,00 DM, auf 4 MB 898,00 DM

Diskettenlaufwerke

Diskettenlaufwerke: 3,5-Zoll- und 5,25-Zoll-Disketten-Laufwerke in vollendeter Qualität. Es werden nur die besten Materialien verwendet. Laufwerksgehäuse mit kratzfester Speziallackierung. 5,25-Zoll-Laufwerk incl. beige Frontblende, 40/80-Track-Umschalter und Software IBM-Atari, anschlußfertig 298,00 DM, Chassis Atari modifiziert 239,00 DM, 1,2-MB-Laufwerk 339,00 DM, 1,44-MB-Laufwerk 279,00 DM, 3,5-Zoll-LW incl. beige Frontblende mit NEC FD 1037 oder TEAC FD 235 anschlußfertig 239,00 DM, Chassis 179,00 DM.

Atari-Zubehör

Auto-Monitor-Switchbox: A.R.S. (Automatic Resolution Selection). Das Programm wird automatisch in der richtigen Auflösung gestartet (nur TOS 1.0 und 1.2). Mit der Auto-Monitor-Switchbox können Sie über die Tastatur zwischen Monochrom und Farbmonitor umschalten oder einen Tastaturreset durchführen. Die mitgelieferte Software ist resetfest. Durch Einbinden der von uns mitgeliefertе-n Routinen Umschaltmöglichkeit ohne RESET. Zusätzlicher BAS und Audio-Ausgang. Auto-Monitor Switchbox 59,90 DM, Auto-Monitor Switchbox Multisync 69,90 DM, weitere Modelle: von 29,90 DM bis 69,90 DM

Video Interface +: ermöglicht die Farbwiedergabe an einem Farbfernseher, Monitor oder Videorecorder mit Videoausgang (mit integrierter Auto-Monitor-Switchbox-Funktion). 159,00 DM

Neu: Echtzeit-Videodigitalisierer in 16 Graustufen (Einlesen von Videosignalen im Computer, kein Standbild erforderlich). 398,00 DM

HF-Modulator: zum Anschluß des Atari ST an jeden gewöhnlichen Farbfernseher. Der Ton wird über den Fernseher übertragen.
189,00 DM, Aufpreis Monitorswitchbox 30,00 DM

Festplattenzubehör: wie SCSI Hostadapter, Einschaltverzögerungen, 1,2 m- DMA-Kabel etc.

STTAST II: ermöglicht den Anschluß einer beliebigen PC-(XT-) Tastatur am ST, umschaltbare Mehrfachbelegung der Tastaturbelegungen, freie Programmierbarkeit von Makros und Generieren von Start-Up-Files (mit AUTO Load), Tastaturreset, unterstützt auch PC Ditto und PC/AT Speed. 149,00 DM Set: PC Tastatur mit Mikroschalter + ST Tast II 298,00 DM

Abgesetzte Tastatur am ST: Tastaturgehäuse mit Spiralkabel, Treiberstufe, Resettaste und Joystickbuchsen eingebaut. Computertyp angeben. 109,00 DM

Towergehäuse: nur Gehäuse oder mit kundenspezifischer Bestückung ab 298,00 DM

RTS Tastaturkappen: ab 89,00 DM

Uhrmodul intern: die Bootsoftware befindet sich auf ROM's im Betriebssystem. Wichtig: Betriebssystem angeben. ROM TOS oder Blitter TOS. 119,00 DM

Floppyswitchbox: ermöglicht den Anschluß von drei Laufwerken am ST. Ausgestattet mit speziellen Treibern für 3,5 und 5,25 Laufwerke. Computertyp angeben. 89,00 DM.

Außerdem: HD-Kid – interne Beschaltung incl. Software zum Anschluß von HD-(3,5"- und 5,25"-Laufwerken am ST. Keine Zusatzschaltung im Laufwerk erforderlich, 59,90 DM

Gehäuse – 3,5" und 5,25"-Gehäuse incl. Montagematerial 29,90 DM, Festplattengehäuse (Mega-Format) 99,00 DM.

SCSI-Hostadapter – orig. ICD-Hostadapter, Advantage+ (mit Uhr) 298,00 DM, Advantage (ohne Uhr) 269,00 DM, Mikro 249,00 DM. Lieferung incl. DMA-Kabel, ICD-Software und Dokumentation.

Einschaltverzögerung – Ermöglicht das gleichzeitige Einschalten von Computer und Festplatte, Einbau erfolgt im Computer durch einfaches Aufstecken, Zeitverzögerung einstellbar, 49,90 DM

Tastaturkabel Mega ST – langes Tastaturkabel (2 m) für den Mega ST 29,90 DM

Eprombrenner – orig. MAXON Junior-Prommer, 229,00 DM, Epromkarte 128KB incl. Gehäuse (Steckmodul), 49,90 DM

Laserinterface – Abschalten des Laserdrucker im Festplattenbetrieb möglich. Lieferung komplett mit Netzteil VDE, GS, 69,00 DM.

MS-DOS-Emulatoren – MS-DOS-Emulator zum internen Einbau. PC Speed 349,00 DM. AT Speed 449,00 DM. Super-Charger MS-DOS-Emulator zum externen Anschluß 729,00 DM.

Hypercache – 16-MHz-Erweiterung mit 16 KB Cache (orig. PRO VME), 550,00 DM. Auch gegen Aufpreis als besonders stromsparende CMOS-Version lieferbar

Fordern Sie unseren Gesamtkatalog an.

Speichererweiterungen
Festplatten
Wechselplatten
Streamer
Diskettenlaufwerke
HD-Diskettenlaufwerke
HD-Kids
Laufwerkschassis
Monitorumschaltboxen
Videointerface
Videodigitalizierer
HF-Modulator
SCSI-Hostadapter
Einschaltverzögerungen
PC-Tastatur an Atari ST
Towersysteme
Towerzubehör
abgesetzte Tastatur am ST
RTS-Tastaturkappen
Echtzeituhren
Floppyswitchbox
Laufwerksgehäuse
Software (PD)
Disketten
Festplattengehäuse
Computerkabel
langes Mega-Tastaturkabel
Eprombrenner
Epromkarten
Laserinterface
PC/AT Speed
Supercharger
Hypercache

Hard & Soft A. Herberg

Obere Münsterstr. 33-35 · 4620 C.-R. · ☎ (0 23 05) 1 80 14 · Fax 3 24 63

1.5e3 auf 1.0e9 Meter vergrößert und die Exzentrizität der Planetenbahn stark verändert. Ansonsten entsprechen die Werte denen des Sonnensystems.

Der Komet wurde, damit er nicht stört, als sehr leicht (1kg) und sehr weit entfernt angenommen. Für die angegebenen Werte ergibt sich eine Planetenbahn wie in Bild 2 gezeigt. Dieser Effekt läßt sich damit erklären, daß dem Planeten nun eine zu einem anderen Abstand gehörende Kraft zugeordnet wird. Der Effekt tritt bei kleinen Abständen im Verhältnis viel stärker auf, so daß der Planet der Sonne nun scheinbar viel näher ist und stärker abgelenkt wird. Es ergibt sich nicht mehr ein einfaches 1/Quadrat(Abstand)-Gesetz, sondern ein viel komplizierteres, bedingt durch die sogenannte Taylorentwicklung.

Das Programm

Nun noch einige abschließende Bemerkungen zum Programm: Die Körper sind durch die Struktur "Koerper" wiedergegeben und werden bei Programmstart automatisch initialisiert. Während des Programmablaufs werden hier die jeweils aktuellen Werte abgelegt. Die Kräfte, die auf einen Körper durch einen anderen wirken, werden durch "get_besch()" berechnet und (ohne Eigenmasse) in "Beschl" fortlaufend aufaddiert. "Beschl" ist vor der ersten Zuweisung zu löschen.

Die Funktion "make_aenderung" berechnet anschließend die Änderungen von Geschwindigkeit und Ort.

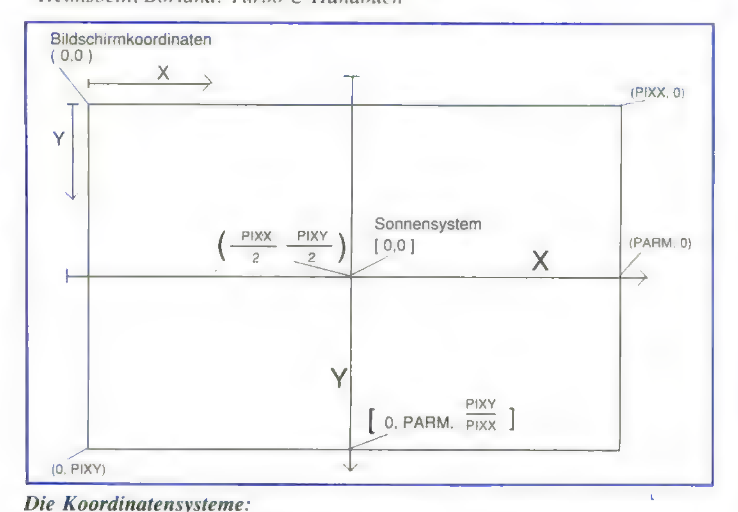
Die Funktion "paint" ist für die Darstellung des Pixels zuständig. Sie sollten hier die entsprechende Funktion Ihres Sprachsystems einfügen bzw. den angegebenen Algorithmus übernehmen, wodurch das Einbinden des VDI überflüssig und der Programmablauf nicht unnötig verzögert wird. Entsprechendes gilt für das Malen des Kreises. Jetzt werden Sie vielleicht sagen, daß das Programm nicht gerade auf Geschwindigkeit und Kürze optimiert ist. Richtig, aber so bleibt es übersichtlich, und die Formeln lassen sich wiederfinden.

Falls Sie das Programm abbrechen wollen, drücken Sie einfach <Shift>. Mit <Control> wird das Bild unter dem Namen "PLANET.PIC" abgespeichert. Das Programm "SHOW" (Listing 2) stellt eine Möglichkeit dar, sich die Bilder noch einmal anzuschauen.

Noch ein Wort zu den Einheiten: Das Programm rechnet nicht in normierten Einheiten, sondern im SI-System, d.h. mit Meter, Sekunde, Kilogramm. Hierdurch ergeben sich zum Teil recht große Zahlen, was vielleicht zu Schwierigkeiten führen kann. Falls Sie umnormieren wollen, dürfen Sie die Gravitationskonstante nicht vergessen.

Albertus Grunwald

Literaturverzeichniss: Gerthsen, Kneser, Vogel: Physik
Heimsoeth, Borland: Turbo C-Handbuch



Die Koordinatensysteme:
Das Programm arbeitet mit zwei Koordinatensystemen

```
 1: /***********************************************
 2: **                                             **
 3: ** Autor: Albertus Grunwald                    **
 4: **                                             **
 5: ** Entwickelt mit Turbo C (9.3.1990)           **
 6: ** (c) MAXON Computer GmbH 1990                **
 7: ***********************************************/
 8:
 9: #include <tos.h>
10: #include <math.h>
11: #include <stdio.h>
12:
13: #define PIXX      640  /* bei anderer Auflösung*/
14: #define PIXY      400  /* entsprechend anpassen*/
15: #define BACK        1
16: #define PAINT       1
17:
18: #define PARM  200.0e9              /* Meter    */
19: #define DT          86400.0*500    /* Sekunden */
20: #define SCHR   0.0e9    /* Schwarzschildradius */
21:
22: typedef struct
23: {          double x,
24:                   y;
25: }vector;
26:
27: struct Koerper
28: {          double Masse;
29:  vector Ort,
30:             Gesch;
31:
32: }Planet,Komet,
33: /*Masse, Ort_X, Ort_Y, Gesch_X, Gesch_Y  */
34:   Sonne ={50.0e23,{0.0,0.0},{0.0,0.0}},
35:   Planet={5.0e23,{110.0e9,0.0},{0.0,-55.0}},
36:   Komet ={1.0e23,{160.9e9,760.0e9},{0.0,-50.0}};
37:
38: vector   Beschl;  /* Beschleunigung ist Kraft
                         ohne Eigenmasse */
39: vector   Loeschen= {0.0, 0.0};
40:
41: double G = 6.672e-11;
42: double Wirkzeit= DT;
43:
44: int *Scr;
45:
46: void get_besch( struct Koerper *von, struct
                 Koerper *auf);
47: void make_aenderung( struct Koerper *aend);
48: void paint( int x, int y );
49: void draw_circle( int x, int y, int r, int f);
50: void cls( void);
51:
52:
53: void main(void)
54: {          int i;
55:
56:  Setcolor(0,0);      /* Farben invertieren */
57:  Setcolor(1,0x777);
58:
59:  Cconout(27);        /* Bildschirm löschen */
60:  Cconout('E');
61:
62:  Scr=Physbase();     /* Setzen des
                            Bildschirmpointers
63:                                 für 'paint' */
64:  for(i=1;i<7;i++)    /* Sonne malen */
65:          draw_circle(PIXX/2, PIXY/2, i, PAINT);
66:
67:  draw_circle(PIXX/2, PIXY/2, PIXY/2-1, PAINT);
68:                      /* Orientierungskreis */
69:
70:  while((i=(int)Kbshift(-1))==0)
71:   {
72:          Beschl= Loeschen;            /* alte
                          Beschleunigung löschen */
73:          get_besch(&Planet,&Komet); /*Änderung
                                     des Kometen */
74:          get_besch(&Sonne,&Komet);
75:          make_aenderung(&Komet);
76:          paint((int)(Komet.Ort.x/PARM * PIXX/2)+
                    PIXX/2,  /* x */
77:                (int)(Komet.Ort.y/PARM * PIXX/2)+
                    PIXY/2); /* y */
78:
79:          Beschl= Loeschen;    /* alte
                     Beschleunigung löschen */    →
```

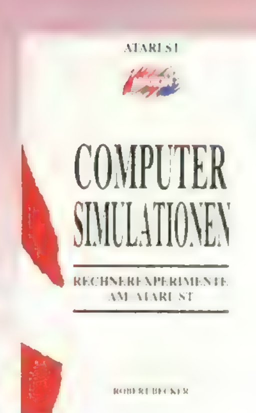
```
80:          get_besch(&Sonne,&Planet); /*Änderung
                                     des Planeten */
81:          get_besch(&Komet,&Planet);
82:          make_aenderung(&Planet);
83:          paint((int)(Planet.Ort.x/PARM * PIXX/2)+
                    PIXX/2,  /* x */
84:               (int)(Planet.Ort.y/PARM * PIXY/2)+
                    PIXY/2); /* y */
85:
86:    }
87:   if((i&0x4)!=0)  /* Bei <CONTROL> wird das Bild
                    gespeichert */
88:    {       int fd;
89:            fd= Fcreate("PLANET.PIC",1);
90:            Fwrite(fd,32000L,Physbase());
91:            Fclose(fd);
92:    }
93:   Setcolor(0,0x777); /*  Farben zurücksetzen */
94:   Setcolor(1,0);
95:  }
96:
97:  /*********  Beschleunigung bestimmen  ***********
98:  *
99:  *  Berechnung der Kraft von Körper 'von' auf
100: *  Körper 'auf' und aufaddieren.
101: *
102: */
103:
104: void get_besch( struct Koerper *von, struct
               Koerper *auf )
105: {          double kraft,x,y,abstand;
106:
107:  x= von->Ort.x - auf->Ort.x;      /* delta_x */
108:  y= von->Ort.y - auf->Ort.y;      /* delta_y */
109:  abstand= sqrt( pow(x,2.0)+pow(y,2.0));
                                      /* Abstand */
110:
111:  abstand-= SCHR;  /* Subtraktion des
                     Schwarzschildradius */
112:
113:  kraft= G * von->Masse / pow( abstand, 3.0);
114:
115:  Beschl.x += kraft * x;  /* Aufaddieren der
                            Kräfte  */
116:  Beschl.y += kraft * y;  /* ( ohne Eigenmasse) */
117:
118:  }
119:
120: /***********   Änderungen vornehmen  ************
121: *
122: *         Berechnung der Wirkung in Ort und
123: *         Geschwindigkeit auf den Körper 'aend'.
124: */
125:
126: void make_aenderung( struct Koerper *aend)
127: {
128:  aend->Gesch.x+= Beschl.x * Wirkzeit;
129:  aend->Gesch.y+= Beschl.y * Wirkzeit;
130:
131:  aend->Ort.x+= aend->Gesch.x * Wirkzeit;
132:  aend->Ort.y+= aend->Gesch.y * Wirkzeit;
133: }
134:
135: /*************  Kreis zeichnen  ****************
136: *
137: *  Zeichnen eines Kreises um das Pixel (x,y) mit
138: *  Radius r und Farbe f nach dem Satz von
139: *  Pythagoras. Es wird nur ein Achtelkreis be-
140: *  rechnet. Der Vollkreis entsteht durch Spie-
                                          geln.
141:                                           */
142:
143: void draw_circle(int x, int y, int r, int f)
144: {
145:  register int i,j;
146:
147:  if (f)
148:         for (i=0; i<=(r+1)*0.75; i++)
149:         {
150:                 j=sqrt((double) r*r - i*i);
151:                 paint(x+i,y+j);
152:                 paint(x+i,y-j);
153:                 paint(x-i,y+j);
154:                 paint(x-i,y-j);
155:                 paint(x+j,y+i);
156:                 paint(x+j,y-i);
157:                 paint(x-j,y+i);
158:                 paint(x-j,y-i);
159:         }
160: }
161:
162: /*************** Pixel setzen  ******************
163: *
164: * Setzen eines Pixel mit den Koordinaten (x,y).
165: * Damit nicht wild in den Speicher geschrieben
166: * wird, testet das Programm die Koordinaten (Cli-
167: * pping). AnschliePend wird das zu setzende Pixel
168: * berechnet */
169:
170: void paint( int x, int y)
171: {        x = x>PIXX-2 || x<1 ?0:x;  /* Clipping */
172:  y = y>PIXY-2 || y<1 ?0:y;
173:
174:  if(x>0 && y>0)
175:     Scr[y*40+x/16] |= (int) (1<<(15 - x & 0xF));
176:   /*^Word^  odern mit einer um ^ verschobenen 1*/
177: }
178:
179: /**********************************************/
180: /**********************************************/
181: /*
182:
183:     ! Ab hier wird nicht mehr compiliert !
184:
185: *************  Daten Komet stört Planet ********/
186:
187: #define PARM  200.0e9                /* Meter    */
188: #define DT        86400.0*500        /* Sekunden */
189: #define SCHR    0.0e9    /* Schwarzschildradius */
190:
191:   Sonne ={50.0e23,{0.0,0.0},{0.0,0.0}},
192:   Planet={5.0e23,{110.0e9,0.0},{0.0,-55.0}},
193:   Komet ={1.0e23,{160.9e9,760.0e9},{0.0,-50.0}};
194:
195: *******************  Daten Merkur  **************
196:
197: #define PARM   80.0e9                /* Meter    */
198: #define DT        86400.0/10         /* Sekunden */
199: #define SCHR   1.0e9     /* Schwarzschildradius */
200:
201:   Sonne ={1.99e30,{0.0,0.0},{0.0,0.0}},
202:   Planet={3.2e23,{50.0e9,0.0},{0.0,-30.0e3}},
203:   Komet ={1.0e00,{1.0e100,0.0},{0.0,0.0}};
204:
205:
206:   ! Ab hier wird wieder compiliert ! */
207:
208: /******************  ENDE  ********************/
```

Listing: Die Simulation der Gravitationskraft

```
 1:  #include <tos.h>
 2:  #include <stdio.h>
 3:
 4:  main(int  argc, char *argv[])
 5:  {
 6:   int fd;
 7:
 8:   Setcolor(0,0);      /* Farben invertieren */
 9:   Setcolor(1,0x777);
10:
11:   if((fd= Fopen(argv[1],0))<0 || Fread(fd,32000L,
                                  Physbase())<0)
12:          printf(" Fehler");
13:
14:   Cconin();
15:
16:   Setcolor(0,0x777); /*  Farben zurücksetzen */
17:   Setcolor(1,0);
18:  }
```

Alles Lesens-Werte

Experimente am Schreibtisch



Durch schnelle Computer kann seit geraumer Zeit die Simulation von Naturvorgängen realistisch vorgeführt werden.
Mit diesem Buch werden Sie in die Lage versetzt, in Ihrem ATARI ST Galaxien kollidieren zu lassen, ohne daß der Himmel einstürzt, oder gar die gewagtesten chemischen Experimente zu riskieren, ohne daß gleich das Haus in die Luft gesprengt wird.
Natürlich darf bei allem Spieltrieb die Theorie der Gesetzmäßigkeiten nicht zu kurz kommen. Ausführlich dokumentierte Listings in GFA-BASIC geben einen Einblick in die programmiertechnische Seite.

Robert Becker
Computersimulationen
337 Seiten und Diskette
DM 59,-
ISBN 3-927065-03-X

Fürs Programmieren unentbehrlich

Die Art und Weise wie benutzererstellte Daten behandelt werden, ob beim Speichern, beim Sortieren oder beim Suchen, ist ein zentrales Thema in der Informatik. Grund genug für Dirk Brockhaus, den verschiedenen Strukturen und Algorithmen auf mehr als 400 Seiten auf den Leib zu rücken.
Angefangen mit den Grundlagen, wie einfachste Strukturelemente, über komplexe Zahlen, sowie verschiedene Methoden für Suchen und Sortieren, bis hin zur Verarbeitung großer Datenmengen umspannt dieses Buch den gesamten Themenbereich.

Dirk Brockhaus
Datenstrukturen
403 Seiten und Diskette
DM 59,-
ISBN 3-927065-02-1

Mehr als ein Buch! Mehr als nur Software!



Mittlerweile zählt Scheibenkleister II bereits zu den Standardwerken für den ST. Die Erfolgsautoren Claus Brod und Anton Stepper beschreiben auf fast 900 Seiten alles, was man über Floppies, Festplatten, CD-ROMs und andere Massenspeicher wissen muß. Das Buch beinhaltet einen großen Kursteil, in dem Themen wie Floppy- und Festplattenprogrammierung (über Betriebssystem und direkte Controllerprogrammierung), Aufzeichnungsverfahren, Anschluß von Fremdlaufwerken und sogar Justierung und Reparaturhinweise von Laufwerken u.v.m. Zusätzlich wird eine Diskette mit 1.2 MB Software mitgeliefert, die keine Wünsche offen läßt:
Track- und Sektormonitor, bootfähige RAM-Disk, Hyperformat bringt bis zu 950 kByte, Festplattentreiber CBHD.SYS (Laden von Accessories von beliebigem Laufwerk) u.v.m.

Brod/Stepper
Scheibenkleister II
872 Seiten
und Diskette
DM 79,-
ISBN 3-927065-00-5

Das Buch zum Handbuch



OMIKRON.BASIC, mittlerweile als ATARI-BASIC bei jedem ST beigefügt, wird in diesem Buch ausführlich beschrieben.
Eine kurze Anleitung für den Neuling führt in die grundlegenden Kennisse der BASIC-Programmierung ein.
Es folgt die Beschreibung der Befehle, Programmierkniffe mit vielen Beispielen, sowie Aufgaben und Lösungen.
Sprites werden erklärt, die Overlaytechnik zeigt das Auslagern langer Programmteile nebst Laden bei Gebrauch.
Was sind Bibliotheken (Libraries)? Das Mitbenutzen fremder Bibliotheksfunktionen, Aufrufe von TOS und GEM, sowie GDOS wird kein Geheimnis bleiben.
Ein weiterer Schwerpunkt ist die Sound- und Grafikprogrammierung unter OMIKRON-Basic, mit Klarheit über Metafiles, IMG-Format usw. Als Spezialität folgt die Beschreibung von internen Multitasking-Befehlen.

Clemens Hoffmann
OMIKRON.BASIC 3.0
355 Seiten
und Diskette
DM 59,-
ISBN 3-927065-01-3



Bestellcoupon MAXON Computer GmbH Schwalbacherstr. 52 6236 Eschborn Tel.: 06196/481811

Name/Vorname

Straße

Ort

Unterschrift

Hiermit bestelle ich:

☐ SCHEIBENKLEISTER II DM 79,00

☐ COMPUTER-SIMULATIONEN DM 59,00

☐ OMIKRON.BASIC 3.0 DM 59,00

☐ DATENSTRUKTUREN DM 59,00

Versandkosten: Inland DM 7,50
Ausland DM 10,00

Auslandbestellungen **nur** gegen Vorauskasse
Nachnahme zuzgl. DM 4,00 Nachnahmegebühr.

☐ Vorauskasse
☐ Nachnahme

Vertrieb in der Schweiz: DTZ Data Trade AG Landstrasse 1 CH-5415 Rieden-Baden Tel.: 01/ 242 80 88 Fax: 01/291 05 07
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3
Vertrieb in Frankreich: AROBACE 2 Rue Piemontesis F-75018 Paris Tel: 1/42235044 Fax 1/42545631



BASEFIND!!!

Was geht im Speicher meines Rechners vor?

Mehrere Dinge mußten zusammenkommen, ehe sich mein von Diskussionen um diverse propagierte 'Standards' (z.B. XARG-ARGV auf USENET [1]) gebeutelter Geist dazu aufraffte, nun selbst einmal einen 'Standard' (tongue in cheek - schmunzeln erlaubt!) vorzuschlagen und dieses kleine Assembler-Programm zu erstellen, um meiner Neugier betreffs im RAM des Rechners lauernder Programme frönen zu können.

Ein bißchen XBRA...

Seit der Vorstellung der sog. XBRA-Konvention (siehe Bild 1) für 'vektorverbiegende' Progamme (vor allem durch Julian 'XBRA' Reschke im Profibuch [2] und in unzähligen (!) Artikeln des bei M&T erscheinenden ST-Magazins, in der ST Computer durch Alex Esser, Claus Brod und andere) habe auch ich selbst dieses Protokoll in die einschlägigen eigenen Programme eingebaut und einige ältere nachträglich damit ausgerüstet. Inzwischen wurde diese neue Konvention von vielen anderen Programmierern positiv aufgenommen und in der Praxis angewandt [3], auch entstanden schon etliche Hilfsprogramme [4], die die Möglichkeiten von XBRA auszunutzen versuchen. Basierend auf diesen Erfahrungen scheint mir eine Reflexion über Sinn und Zweck der ganzen Angelegenheit vielleicht nicht ganz unangebracht.

Klinkenputzer

Die Idee einer nach einem festgelegten Schema verketteten Liste von Vektoren hat viel für sich... Die Sache hat jedoch den Haken, daß die beiden Hauptvorteile (nach [2]), die man mit der Einführung dieser Konvention erreichen wollte, so einfach nicht unbedingt zu verwirklichen sind, und zwar 1., daß ein Programm leicht feststellen kann, ob es bereits installiert ist, und 2., daß ein Programm sich selbst (oder nur die entsprechende Routine) leicht aus der Liste wieder entfernen kann. (Das Wörtchen 'leicht' ist in diesem Zusammenhang allerdings wenig aussagekräftig, wie wir gleich sehen werden.)

Zu 1.: Das war noch nie ein Problem. Gegen die hier implizierte Methode spricht jedoch die Tatsache, daß schon ein einziges Programm, das sich nicht an die Konvention hält, die Liste unterbricht und den vor ihm liegenden Rest 'unsichtbar' macht. Zu 2. läßt sich sagen, daß es schon immer 'leicht' möglich (und auch üblich) war, programmeigene Routinen z.B. bei Terminierung wieder auszuklinken - im Sinne einer Wiederherstellung des bei Programmstart vorgefundenen Zustands. XBRA wird erst dann interessant, wollte man von später gestarteten (fremden) Prozessen verursachte Zustände erhalten. Dies wiederum würde nur bei konsequenter, d.h. ausschließlicher Anwendung von XBRA erleichtert, worauf man sich jedoch prinzipiell (!) *nicht* verlassen darf: Erleichterung, ade! Worin also liegt der Nutzen? Das XBRA-Protokoll wäre vielleicht noch von Interesse, wollte man programmfremde Vektoren ausklinken (wozu? [5]) oder einfach nur betrachten (interessant!), und das scheitert dann auch wieder an dem zu 1. genannten Vorbehalt [6] (Bild 2).

Das 'Ausklinken' residenter Utilities, ob mit oder ohne Terminierung, hinterläßt ja immer einen mehr oder weniger zerstückelten Speicher (nicht zu nutzende freigegebene Bereiche oder 'Programmleichen'). Aus einer Shell, die z.B. an einer Reihe von Vektoren herumbiegt, residente Programme zu starten, die ihrerseits die gleichen und/oder andere Vektoren manipulieren, dann die Shell (mit Rücksetzen oder Ausklinken der benutzten Vektoren) zu verlassen, um mit anderen Programmen (womöglich nach gleichem Muster!) weiterzuarbeiten, wäre eine der Konzeption und den Möglichkeiten des Betriebssystems wenig angemessene, äußerst ineffiziente Nutzung des Anwenderspeichers:

Vektorverbiegende residente Utilities gehören in den AUTO-Ordner oder sollten als Accessories ausgeführt sein! Für eine ausführliche Diskussion dieser Thematik siehe den grundlegenden Beitrag von Alex Esser in [10], zur Speicher- und Programmverwaltung des TOS auch [7].

Eine heikle Angelegenheit also, dieses Ausklinken von tiefer liegenden Programmen, und wohl hauptsächlich angestrebt, um ohne Reset auskommen zu können. Der Reset (Warmstart) ist nämlich die vom Betriebssystem vorgesehene Kur, um das ganze Gewusel von 'verbogenen' Vektoren wieder in einen definierten Ausgangszustand zu bringen, die systeminternen Organisationsstrukturen zu aktualisieren und gleichzeitig den Speicher aufzuräumen. Ein Programm, das ganz bewußt diesen Weg geht, ist z.B. das HDX.PRG von ATARI. Alternativ könnte man - nach Verlassen des Programms - des Anwenders Abenteuerlust mit 'merkwürdigen Effekten' kitzeln und ihn einige Zeit bis zum 'unerklärlichen Absturz' weiterarbeiten lassen [8]! Zuweilen hilft da nur noch der Kaltstart, als Radikalkur par excellence!

Auf der Suche...

Das Suchen nach dem eigenen Programm, um z.B. Mehrfachinstallation zu verhindern, muß also aus Sicherheitsgründen auf konventionelle Weise erfolgen, und dazu könnte man natürlich auch die XBRA-Struktur benutzen, wenn sie denn ohnehin im Programm vorgesehen ist (Bild 3). Für Analyse- programme könnte das Vorhandensein von XBRA dazu dienen, die Belegung des Speichers mit residenten Routinen durchsichtiger zu machen.

Ebenfalls ein Suchproblem hatte ich bei der Programmierung eines sehr hoch optimierten Accessories, als ich bestimmte Zustände in dessen Basepage testen mußte. Ein 'normales' Programm läßt sich dazu aus dem Debugger starten, aber wie findet man die Basepage eines Accessories im Speicher? Nachdem ich es dann endlich mittels Suche nach einem darin enthaltenen ASCII-String lokalisiert und mich wieder einmal mit der Struktur der Basepage (Bild 4) vertraut gemacht hatte, sann ich auf einen eleganteren Weg. Et voilà, hier sind das Programm BASEFIND.TOS und mein Vorschlag für das XNAM-Protokoll (Bild 6).

Die Basepage

Das herausragende Kennzeichen der Basepage, auch PD (Program Descriptor: enthält alle wichtigen Verwaltungsinformationen des Programms) genannt, ist,

```
...          irgendwo im Programm

xb_magic: dc.l       'XBRA'      ;XBRA-Kennung
xb_id:    dc.l       'abxy'      ;XBRA-ID, 4 m. oder w. sinnvolle ASCII Z.
xb_ovec:  dc.l       0           ;hier vorgefundenen Vektor ablegen

new_vec:  ...                    ;Einsprungstelle für neue Routine

*                    ...          hier folgt nun, was gemacht werden soll,
*                    ...          und so endet es meist:

exit:  move.l xb_ovec(pc),a0      ;alten Vektor laden
       jmp    (a0)                ;und dahin springen

* Wenn die Routine ohne Reset entfernt werden soll:

* 1. Normalfall: 'xb_ovec' wieder in die Sysvar schreiben und terminieren.

* 2. XBRA: Eigenen 'xb_ovec' für 'xb_ovec' des Vorgängers einsetzen oder in
* die Sysvar (wenn diese eigenen 'new_vec' enthält) und terminieren.

* 3. Ausklinken in unterbrochener Liste: 'xb_id' z.B. mit 'KILL' ungültig
* machen, folgenden Code nach 'new_vec' kopieren, diese Struktur resident
* halten und terminieren.

rm_vec: move.l -6(pc),a0          ;alten Vektor laden
        jmp    (a0)               ;und dahin springen

rm_vec_h: dc.w $207A,$FFFA,$4ED0  ;obiger Code als Hexdump

* Wenn dann über 50 ausgeklinkte Vektoren in der Liste sind, würde ich (nur
* aus ästhetischen Gründen!) vielleicht doch lieber einen Reset empfehlen!
```

Bild 1: Die XBRA-Struktur, zur Abwechslung einmal in Assembler-Notation

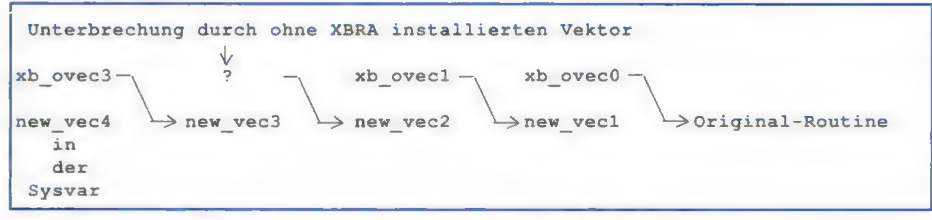


Bild 2: Durch das XBRA-Protokoll verkettete Vektoren-Liste

```
* Irgendwo im residenten Teil des Programms
*                    steht folgendes vor der zu installierenden Routine:

xb_magic: dc.l       'XBRA'        ;XBRA-Kennung
xb_id:    dc.l       'ABCD'        ;ID
xb_ovec:  dc.l       0             ;Platz für den alten Vektor
new_vec:                           ;hier geht's los
*         ...

* Suchroutine des Installationsteils (wegen Sysvarmanipulation im Super!):

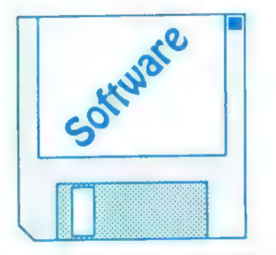
          lea       start-$100(pc),a0    ;Programmstart - Länge des PD
          move.l    a0,d1                ;für Zählerberechnung
          movea.l   $432,a0              ;membot -> Suchanfang
          sub.l     a0,d1                ;so viel Byte durchsuchen
search:   addq      #2,a0                ;nur gerade Adressen
          subq.l    #2,d1                ;Zähler dekrementieren
          bmi.s     action               ;Programm installieren!

          move.l    a0,a2                ;Suchadresse
          lea       xb_magic(pc),a1      ;Adresse XBRA-Kennung
          cmpm.l    (a1)+,(a2)+          ;xb_magic?
          bne       search               ;weitersuchen

          cmpm.l    (a1)+,(a2)+          ;xb_id?
          bne       search               ;weitersuchen

          tst.l     (a2)                 ;= 0 -> nicht installiert
          beq       search               ;weitersuchen

* der folgende Test soll z.B. vor ASCII schützen und kann entfallen, wenn
* 'xb_ovec' - wie gezeigt - sinnvollerweise mit Null vorbelegt ist:
                                                                      →
```



Neuheiten

ST-Modular

Modula-2 für ATARI ST

Das MSM2-System ist ein vollständiges, professionelles Modula-2-Entwicklungssystem:

- Optimierter Linker zur Erzeugung kompakter Programme und Accessories
- Inline-Assembler mit Zugriffsmöglichkeit auf Modula-Variablen
- Projekt-Unterstützung
- Debugging auf Modula- und Maschinencode-Ebene (auch beides gemischt)
- Komplettes Monitorprogramm mit Tracer ist im System integriert
- Umfangreiche Modulbibliothek mit Quelltexten

Die Benutzerfreundlichkeit des MSM2-Systems ist unübertroffen:

- Es handelt sich um ein integriertes System, das alle zur Programmentwicklung nötigen Komponenten ständig im Speicher hält. Ladezeiten, wie sie bei shell-gesteuerten Systemen üblich sind, treten nicht mehr auf. Dabei findet das vollständig in Assembler programmierte System mit seinem Codevolumen von nur 80 KB auch im kleinsten ATARI ST ausreichend Platz
- Durch die Möglichkeit, alle benötigten Module (bis 48) resident im Speicher zu halten, treten bei Übersetzung und Start eines Programms keine Modul-Ladezeiten auf
- Der Compiler arbeitet extrem schnell: Er übersetzt im Schnitt 20000 Zeilen pro Minute. Er ist zudem direkt vom Editor aus mit nur einem Tastendruck aufrufbar und ruft seinerseits beim Antreffen syntaktischer Fehler den Editor auf

Aus diesen Faktoren resultieren geradezu traumhafte Turn-Around-Zeiten, die selbst bei sehr großen Programmen im Bereich weniger Sekunden liegen.

Das Handbuch (460 Seiten) enthält neben einer verständlichen System- und Sprachbeschreibung ein Tutor-Kapitel, welches anhand einer konkreten Programmentwicklung in die Systembedienung einführt.

Unverbindlich empfohlener Verkaufspreis **DM 298,-**

ST-GEO

Geographie-Software

Systemanforderungen:
ST-GEO ist auf allen ATARI-ST Modellen mit monochromem Monitor (SM 124) und mindestens einem Megabyte Speicher lauffähig, unabhängig davon, welche TOS-Version verwendet wird.

Leistungsmerkmale:

- Voll GEM-unterstützte Bedienung (Mausbedienung, Fenstertechnik, Pulldown-Menu...)
- Viele Funktionen auch über Tastatur anwählbar
- Hilfstexte werden jederzeit angezeigt
- Umfangreicher Abfragemodus mit graphischer Auswertung
- Suchmodus für unbekannte Länder, Bezirke, Kantone oder Städte
- Landkarten von der Erde, allen Kontinenten, von Deutschland (BRD/DDR) und der Schweiz
- Staatswappen für 166 Länder, 13 Bundesländer und 26 Kantone
- Folgende Daten sind von allen 166 Ländern verfügbar:
  - Hauptstadt - Sprache - Währung
  - Staatsform - Fläche - Einwohner
  - Einwohner/km² - Religion - Zeitzone
- Folgende Daten sind von 26 Bundesländern/Bezirken von Deutschland verfügbar:
  - Hauptstadt - Sprache - Fläche
  - Einwohner - Einwohner/km²
- Folgende Daten sind von 26 Kantonen der Schweiz verfügbar:
  - Hauptort - Sprache - Beitritt zur Schweiz
  - Fläche - Einwohner - Einwohner/km²
- Gespeicherte Daten können geändert bzw. aktualisiert werden
- Ausdruck der Landkarten oder der Daten der einzelnen Länder
- Druckertreiber für 9-Nadler (FX-kompatible) und HP-Laserjet
- Daten über wichtige Gebirgszüge und Gewässer
- Geographische Rekorde der einzelnen Kontinente/Länder
- Daten von politischen Organisationen wie UNO, EG, OPEC, OAU...
- dreidimensionale Darstellung der Erdkugel aus beliebiger Position im Weltall
- Animation der Erdkugel

Unverbindlich empfohlener Verkaufspreis **DM 98,-**

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

BESTELL-COUPON

Bitte senden Sie mir: ____ ST-GEO à 98,- DM (zzgl. Versandkosten DM 6,-
____ ST-Modular à 298,- DM Ausland DM 10,-)

Name, Vorname ______

Straße, Hausnr. ______

PLZ, Ort ______

(Oder benutzen Sie die eingeheftete Bestellkarte)

In der Schweiz:
Data Trade AG
Landstr. 1
CH - 5415 Rieden-Baden

daß sie als erstes die Adresse ihrer selbst enthält. Wir brauchen also den relevanten Bereich des Speichers nur nach Adressen zu durchsuchen, die sich selbst enthalten, und schon haben wir alle Basepages lokalisiert. Die Wahrscheinlichkeit, daß uns einmal der Zufall einen Streich spielen könnte, wir also keinen gültigen PD vor uns hätten, ist selbst bei 4 MB mit einem Mischmasch von ASCII und Code vollgestopften Speicher so gering, daß wir diesen Fall guten Gewissens vernachlässigen können. Die Mathematiker unter den Lesern könnten es sich ja ruhig einmal ausrechnen!

Betrachtet man die Bildschirmausgabe von BASEFIND (Bild 5), sind in der ersten Spalte die im Speicher eingerichteten PDs zu sehen, und zwar beginnend mit den niedrigen Adressen. In der zweiten Spalte habe ich die ursprünglich (beim Start) vorhandene Länge der zu diesen PDs gehörigen Programme auflisten lassen. Das sind lediglich die aufaddierten Werte von *p_textlen*, *p_datalen* und *p_bsslen*, es bedeutet nicht notwendigerweise den Abstand zum nächsten PD, denn kluge Programmierer werfen nach erfolgter Installation nicht mehr benötigten Code etc. fort und halten nur das wirklich Gebrauchte resident. Hier gewinnen wir nun einen schönen Einblick in die Speicherverwaltung und -aufteilung unseres Rechners und können auch das Verhalten uns sonst unbekannter Programme kontrollieren.

Eltern und Kinder

In der dritten Kolumne sind die Zeiger auf die Elternprozesse eingetragen, d.h. derjenigen Prozesse, aus denen heraus die zu den jeweiligen PDs gehörenden Programme gestartet wurden. Das bringt nun wieder interessante Einsichten: Der erste angezeigte PD, den wir (nur) auf diese Weise lokalisieren können (Routine *'preloop'*), gehört zum sog. Urprozeß, dessen (zum größten Teil leere) Basepage während der GEMDOS-Initialisierung eingerichtet wird. Wird der ATARI-Harddisk-Treiber von der Platte gebootet, reloziert ihn das Boot-Ladeprogramm und springt ihn dann direkt an: Er ist dann kein eigenständiger Prozeß, hat somit auch keinen PD und gehört deshalb zum Urprozeß. Kind eben dieses Urprozesses ist meist das sog. AUTOEXEC.PRG, dessen Kinder wiederum sind die AUTO-Ordner-Programme und schließlich auch der (oder das) AES-Desktop. Dann folgen etliche Programme ohne Eltern (p_parent=0). Die müssen natürlich auch mal gestartet worden sein, und zwar vom AES. Für GEMDOS jedoch, das im Betriebssystem die Speicherverwaltung und -zuteilung

```
        tst.b      (a2)                  ;= 0 -> sinnvolle Adresse
        bne        search                ;weitersuchen

        btst       #0,3(a2)              ;gerade Adresse?
        bne        search                ;weitersuchen

* man könnte natürlich auch noch testen, ob der Vektor in einen sinnvollen
* Bereich zeigt, aber auch das erübrigt sich meist...

no_inst:                                 ;nicht installieren, schon da!
*          ...
action:                                  ;Programm installieren
*          ...
```

Bild 3: Programminstallationsidentifikationsbeispiel (was'n Wort!)

Offset	Länge	Name	Beschreibung
00	4	p_lowtpa	Basisadresse der TPA - die Basepage selbst
04	4	p_hitpa	Ende der TPA + 1
08	4	p_tbase	Basisadresse des Textsegments (Code)
0C	4	p_tlen	Länge (in Byte) des Textsegments (Code)
10	4	p_dbase	Basisadresse des Datasegments
14	4	p_dlen	Länge (in Byte) des Datasegments
18	4	p_bbase	Basisadresse des BSS-Segments
1C	4	p_blen	Länge (in Byte) des BSS-Segments
24	4	p_parent	Zeiger auf die Parent-Basepage
2C	4	p_env	Zeiger auf Environment-String
80	80	p_cmdlin	Command line image
			Nicht genannte Felder sind reserviert!

Bild 4: Die Basepage-Struktur nach dem 'ATARI GEMDOS Reference Manual' [9]

```
basepage  prog_len  p_parent  XNAM  XBRA  _old_vec  _new_vec BASEFIND (c)br

00005FB2  00000000  00000000        CBHD  00FC173C  0000A8B2
                                    CBHD  00FC1A24  0000A8CE
                                    CBHD  00FC18EC  0000A8EC
0000CC90  00000000  00005FB2
0000CD94  000007A0  0000CC90        _OOL  00FC9332  0000D128
0000D678  000003BC  0000CC90  CACH
00024D0C  0000032E  0000CC90  SCRN
00024FAE  00001ED8  0000CC90        AgKO  0000A8B2  000251C6
                                    AgKO  0000A8EC  000251F2
                                    AgKO  0000A8CE  00025218
00026F8A  0000036C  0000CC90        VDOG  000251C6  0002709A
00027348  00000000  0000CC90
0002BAE0  00000F6C  00000000
0002D40C  0000328A  00000000
00031056  000000DA  00000000  FREE
00031AF0  0000BA6E  00000000
0003DF1E  000006B8  00000000        BSPL  00FC07CA  0003E23C
                                    HSPL  00FC32F6  0003E29A
0003EF96  00002D3A  00000000
000451DE  0000DF7A  00027348  SHEL
000579E0  00015B5E  000451DE
0007083E  0006F680  55AA0007
000776DE  00000382  000579E0  BASE  Obviously the program BASEFIND itself!

                                    WINZ  0002709A  00184A04
                                    WINZ  00025218  00184A28
                                    WINZ  000251F2  00184B58

active vbl_slots:   00FD0AE6
                    00024E18  SCRN

magic_pk: 00184600
```

Bild 5: Beispiel einer Bildschirmausgabe des Programms BASEFIND.TOS

besorgt, sind sie Waisenkinder: die Accessories. Das darauf folgende Programm wurde aus dem AES-Desktop gestartet. Hier ist natürlich auch wieder eine verkettete Folge von Prozessen denkbar (sog. spawning). Der letzte angezeigte PD gehört normalerweise zum gerade laufenden (aktuellen) Programm, das wäre eben BASEFIND.TOS, und ich habe es sinnigerweise mit der XNAM_ID 'BASE' ausgestattet, was man in der folgenden Spalte auch sehen kann.

Ich sagte gerade bewußt 'normalerweise', denn das Betriebssystem (hier: GEMDOS) besorgt sich immer das größte zusammenhängende Stück Speicher und lädt die Programme dann an dessen Anfang. Wenn die sich 'normal' verhalten, 'shrinken' und (mit einem Teil) resident bleiben, wird so ein Programm über das andere geladen, und alle liegen hübsch in der historischen Reihenfolge von unten nach oben hintereinander im Speicher. Das führt u.a. zu einer guten Ausnutzung des vorhandenen RAMs. Es ist nun aber durchaus möglich, durch mehrfaches, geschicktes Allozieren und Freigeben von Speicherplatz ein Programm z.B. ganz oben im Speicher zu plazieren. Das mag allerdings nur für sehr spezielle Anwendungen sinnvoll sein, aus diesem Grunde wird die Suche nach weiteren PDs etc. auch nach der Entdeckung der Basepage des eigenen Programms bis zum Speicherende fortgesetzt. Das kann bei 4 MB Speicher schon ca. 40 Sekunden dauern: Immerhin werden fast zwei Millionen Adressen vielfach überprüft, also bitte nicht ungeduldig werden! Wem das zuviel ist, der kann die mit (*) bezeichnete Zeile 131 auskommentieren. Dann entgehen ihm aber auch evtl. sehr hoch plazierte RAM-Disk-Treiber etc..

...ein wenig XNAM...

Die vierte Spalte enthält dann - so vorhanden - die Kennungen nach dem XNAM-Protokoll. Es ist allerdings ein noch sehr junger 'Standard' (au!), z.Z. gerade mal eben von mir und einigen Freunden angewandt und propagiert! Denn nichts sollte den geneigten Leser abhalten, sich dieses vorzüglichen Werkzeugs zu bedienen, hat es doch viele Vorteile und leistet vor allem, was es verspricht! Auch braucht man sich keineswegs registrieren zu lassen, und die Reservierung von Kürzeln halte ich ebenfalls für durchaus unnötig: Wenn der Vorrat an sinnvollen Kombinationen knapp werden sollte, haben wir ja noch YNAM, ZNAM etc.pp. (ähnliches schlage ich übrigens für XBRA vor, wo's allmählich eng zu werden droht, und reserviere - grapsch! - gleich mal YBRA und ZBRA für mich! Die Hybris auf dem Zebra, auweia!)

...und nochmal XBRA

Nachdem das mehr oder weniger erfolgreiche Suchen nach der XNAM-Struktur nun erledigt ist, fahren wir fort und suchen innerhalb des Bereichs jeder Basepage nach evtl. installierten XBRA-Strukturen. Der Bereich eines PD geht dabei bis zum nächsten PD in Richtung aufsteigender

```
* Das nach dem XNAM-Protokoll eingerichtete Programm beginnt so:

start:     bra.s     begin      ;Sprung auf eigentlichen Programmanfang
*start:    bra.w     begin      ;wahlweise bra.w
xn_magic:  dc.l      'XNAM'     ;XNAM-Kennung
xn_id:     dc.l      'ABCD'     ;XNAM-ID, 4 beliebige ASCII-Zeichen
*          ...                  ;Platz für weitere Strukturen
begin:     ...                  ;hier geht's los

* Für Freunde des 'jmp'-Befehls am Programmbeginn (ich gehöre nicht dazu)
* könnte man natürlich noch einen Offset von 6 vorsehen. Vielleicht sollte
* man eine Konferenz einberufen und sich auf einen festen Offset einigen?!

* Wenn man viel Platz verschwenden will, wäre auch dies möglich:

start:     bra.w     begin      ;Sprung auf eigentlichen Programmanfang
xn_magic:  dc.l      'xnam'     ;xnam-Protokoll-Kennung
xn_id:     dc.b '12345678.123'  ;xnam-ID, der komplette GEMDOS-Dateiname,
*                                12 Byte, evtl. mit Leerzeichen aufgefüllt
*          ...                  ;Platz für weitere Strukturen
begin:     ...                  ;hier geht's los
```

Bild 6: Die XNAM-Struktur in Assembler-Notation

```
* Direkt vor der Einstiegsadresse der zu installierenden Slot-Routine liegt
* die XNAM-Struktur:

xn_magic:  dc.l     'XNAM'     ;XNAM-Kennung
xn_id:     dc.l     'ABCD'     ;XNAM-ID, 4 beliebige ASCII-Zeichen
slot:      ...                 ;Code
```

Bild 7: XNAM als Kennung für VBI-Slot-Installation

```
 1:  * ----------------------------------------------
 2:  * basefind.s      find & display basepage
                       locations in memory
 3:  * find XNAM, XBRA (with vecs), vbl-slots, magic
                       packs, GEMDOS i/o
 4:  * ----------------------------------------------
 5:  * Register: a3 Basepage; a4 memtop; a5 XBRA-
             Suchlauf; d7 flag; d3,d4 prt_hex
 6:  *           a4,d5 slotties;
 7:  * ----------------------------------------------
 8:          bra.s      start
 9:  * ----------------------------------------------
10:  xnam:   dc.l       'XNAM'        ;bei Offset 2 o.
                                       4 vom Textbeginn
11:  base:   dc.l       'BASE'        ;steht diese Prg
                                       -Kennung
12:  * ----------------------------------------------
13:  start:  dc.w       $A000         ;line_a init
14:          cmpi       #79,-$2C(a0)  ;v_cel_mx: mini-
                                       mum #columns=80
15:          blt        sorry         ;schade!
16:          cmpi       #24,-$2A(a0)  ;v_cel_my: mini-
                                       mum #lines=25
17:          blt        sorry         ;traurig!
18:
19:          lea        title(pc),a0  ;Titelzeile
20:          bsr        conws         ;ausgeben
21:
22:          pea        memtop(pc)    ;Endadresse
23:          move       #$26,-(sp)    ;Supexec
24:          trap       #14           ;XBIOS
25:          addq       #6,sp
26:
27:          movea.l    d0,a4         ;Endadresse
28:          movea.l    #$800,a3      ;Startadresse
29:
30:  preloop: addq      #2,a3         ;nur gerade Adr.
31:          cmpa.l     (a3),a3       ;enthält Adresse
                                       sich selbst?
32:          bne        preloop       ;weiter
33:          move.l     $24(a3),a3    ;p_parent = neue
                                       Startadresse
34:
35:          move.l     #$600,a5      ;für XBRA-Suche
36:          move.l     a4,a6         ;retten
37:          move.l     a3,a4         ;Bereichsende
38:          moveq      #-1,d6        ;flag für diese
39:
```

→

Adressen, und ist nicht etwa an der ursprünglichen Programmlänge orientiert; die Gründe dafür wurden bereits erwähnt. Zur eindeutigen Feststellung, ob es sich auch wirklich um installierte XBRA-Strukturen handelt, und nicht etwa nur um eine der vielen im Speicher befindlichen Diskussionen über dieses Thema oder schlicht um ein im Cache stehendes Programm, sind allerdings noch einige Plausibilitätsprüfungen nötig. Das Problem ist nämlich, daß die Kennung 'XBRA' kein echtes 'magic' (d.h. mindestens zum Teil nicht ASCII) ist. Daher gestaltet sich diese Suche etwas komplizierter, als man dachte. Dennoch ist es unter unglücklichen Umständen möglich, daß 'ASCII-Mist' ausgegeben wird. Ferner werden auch die XBRAs nicht erkannt, bei denen *xb_ovec* nach TOS-Manier im obersten Byte die Exception-Nummer enthalten sollte. Ich hoffe, daß aus den Kommentaren im Quelltext deutlich genug hervorgeht, wie's gemacht wird.

Das Erfreuliche bei der hier angewandten Methode ist natürlich, daß auch die sonst durch andere Programme verdeckten und dadurch für die reinen Vektorsucher [4] unsichtbaren XBRA-Strukturen ans Tageslicht geholt werden: Dem ermatteten Programmierer entringt sich ein lustvolles Stöhnen: "Endlich! Immer hat diese blöde RAM-Disk mein XBRA verdeckt!" Tja, BASEFIND bleibt eben (fast) nichts verborgen.

So, XBRA wäre damit endlich abgefackelt, wurde ja auch langsam Zeit! "Was bleibt noch Sinnvolles zu tun, zumal das kByte noch nicht voll ist?" fragte ich mich grübelnd. "Die restlichen aktiven Strukturen aufdecken, als da sind die in den '*VBI-Slots*' hängenden (oder vielleicht besser: steckenden?) Routinen und natürlich die sog. 'magic packs'".

Steckplätze...

Die VBI- oder auch VBI-Slot-Routinen werden, sofern sie nicht gerade gesperrt sind, bei jedem Vertical Blank oder Bildrücklauf einmal ausgeführt, und wenn man ihre Adressen mit der Liste der PDs vergleicht, ist sehr schön zu sehen, von welchem Programm sie installiert wurden. Defaultmäßig sind 8 Slots eingerichtet, die Liste kann aber beliebig verlängert oder auch verkürzt werden. Es werden nur die belegten Slots angezeigt. Das AES braucht, wenn es aktiv ist, immer das erste Slot, was man auch hier beobachten kann. In den Slots steckende Routinen lassen sich übrigens sehr einfach deaktivieren, indem man dort eine lange Null hineinschreibt. Deshalb ist, im Gegensatz zur Meinung einiger Autoren (z.B. JXR),

```
 40:            pea     next_2(pc)    ;Teilroutine
 41:            move    #$26,-(sp)    ;Supexec
 42:            trap    #14           ;XBIOS
 43:            addq    #6,sp
 44:
 45:            moveq   #0,d6         ;flag löschen
 46:            move.l  a6,a4         ;alte Endadresse
 47:            bra.s   output        ;zur Ausgabe
 48:
 49: loop:      moveq   #0,d7         ;flag löschen
 50:            addq    #2,a3         ;nur gerade Adr.
 51:            cmpa.l  (a3),a3       ;enthält Adresse
                                       sich selbst?
 52:            beq.s   output        ;dann verarbeitn
 53:            cmpa.l  a3,a4         ;Endadresse?
 54:            beq     task_2        ;fertig
 55:            bra     loop          ;n. Basepage
 56:
 57: output:    move.l  a3,d3         ;Basepage-Adr.
 58:            bsr     prt_hex       ;ausgeben
 59:            bsr     space         ;Zwischenraum
 60:            move.l  $C(a3),d3     ;p_tlen
 61:            add.l   $14(a3),d3    ;p_dlen
 62:            add.l   $1C(a3),d3    ;p_blen addieren
 63:            bsr     prt_hex       ;und ausgeben
 64:            bsr     space
              ;Zwischenraum
 65:            move.l  $24(a3),d3    ;p_parent
 66:            bsr     prt_hex       ;und ausgeben
 67:            lea     $102(a3),a1   ;Offset 'XNAM'
 68:            lea     xnam(pc),a0   ;was drin steh'n
                                       soll
 69:            move.l  (a1),d0       ;gucken
 70:            cmp.l   (a0),d0       ;steht's dort?
 71:            beq.s   right_on      ;ok
 72:            addq    #2,a1         ;um 2 erhöhen
 73:            move.l  (a1),d0       ;nachschauen
 74:            cmp.l   (a0),d0       ;steht's dort?
 75:            bne.s   no_name       ;leider nicht
 76:
 77: right_on:  lea     write(pc),a0  ;Leerstring
 78:            move.l  $4(a1),d3     ;merken
 79:            move.l  d3,2(a0)      ;Namen dorthin
 80:            bsr     conws         ;und ausgeben
 81:            cmp.l   base(pc),d3   ;das prg höchst-
                                       selbst?
 82:            beq     special       ;zur Spezialausg
 83:
 84:            bra.s   next_1        ;weiter
 85:
 86: no_name:   lea     empty(pc),a0  ;6 * Space
 87:            bsr     conws         ;ausgeben
 88: next_1:    lea     $100(a3),a5   ;Suchanfang
 89: next_2:    addq    #2,a5         ;nur gerade Adr.
 90:            cmp.l   (a5),a5       ;schon n. BP?
 91:            beq.s   thats_it      ;Suche beenden
 92:            cmpa.l  a5,a4         ;Endadresse?
 93:            beq.s   thats_it      ;fertig
 94:            move.l  xbra(pc),d0   ;Kennung
 95:            cmp.l   (a5),d0       ;steht's da?
 96:            bne     next_2        ;weitersuchen
 97:
 98:            lea     4(a5),a0      ;xb_id
 99:            moveq   #3,d1         ;4 Byte
100: test_a:    cmpi.b  #' ',(a0)+    ;ASCII?
101:            bls     next_2        ;nein -> weiter
102:            dbf     d1,test_a     ;Langwort
103:
104:            move.l  8(a5),d3      ;xb_oldvec
105:            beq.s   next_2        ;war nichts!
106:            tst.b   8(a5)         ;sinnvolle Adr?
107:            bne.s   next_2        ;wieder nichts!
108:            btst    #0,$B(a5)     ;Adresse gerade?
109:            bne.s   next_2        ;total entnervt!
110:
111:            tst     d7            ;flag testen
112:            beq.s   skip_1        ;erstes Mal
113:
114:            lea     posit(pc),a0  ;Cursor neu
                                       positionieren
115:            bsr.s   conws         ;und ausgeben
116: skip_1:    lea     write(pc),a0  ;Leerstring
117:            move.l  4(a5),2(a0)   ;Namen dorthin
118:            bsr.s   conws         ;und ausgeben
119:            bsr.s   space         ;Zwischenraum
120:            bsr     prt_hex       ;xb_oldvec ausg.
```

→

XBRA hier gänzlich unangebracht. (xb_ovec würde dann den hochinteressanten Wert Null enthalten!) Doch kann hier die XNAM-Kennung, direkt vor den Einstieg gelegt, die Installationsidentifikation erleichtern (Bild 7). - Die ganze Routine muß übrigens im Super abgearbeitet werden, weil die Slot-Liste defaultmäßig so weit unten liegt.

...und magische Päckchen

Damit kämen wir schließlich zur Anzeige der evtl. vorhandenen *'magic packs'* (ich muß da immer an *'sixpack'* denken!). Das sind RAM-residente Programme, die nach einem speziellen und äußerst 'magischen' Ritual eingerichtet sind und nach den Boot-Programmen, aber vor den AUTO-Ordner-Programmen abgearbeitet werden. Meist werden sie noch zusäzlich resetfest gemacht und für RAM-Disks, zur Vektormanipulation und zum Patchen des Betriebssystems gebraucht, es gibt auch Viren, die diese Methode benutzen. Um diese Programme zu deaktivieren, genügt es, das 'magic' ungültig zu machen. Einzelheiten in [2;7]. Wer will, nehme einfach den Debugger und schaue sich die entsprechende Stelle im TOS an (z.B. bei Offset $E56 im Blitter-TOS; die Buchautoren haben auch nichts anderes getan, d.h. dieses Feature ist nicht offiziell dokumentiert) oder mangels Debuggers die entsprechende Stelle im Programm, denn sie ist bis auf unwesentliche Änderungen direkt dem TOS entnommen. Dort wird natürlich so ein Programm ausgeführt und nicht, wie hier, nur gesucht und angezeigt. Auch hier ist wieder Super angesagt, weil bis einschließlich $600 gesucht werden muß.

Alles gefunden?

Haben wir damit nun alle möglichen aktiven Programmstrukturen im Speicher entdeckt und sichtbar gemacht? Sofern es sich um 'normal' gestartete Prozesse handelt, haben sie auch einen PD, und der läßt sich auch auf die beschriebene Weise finden. Belegte VBI-Slots und 'magische Päckchen' kann man ebenfalls leicht lokalisieren. Nicht mit XBRA installierte Vektoren bleiben dem Programm natürlich verborgen, von ihnen könnte zumindest jeweils der erste einer Kette von den anders operierenden Vektorsuchprogrammen gefunden werden. Ferner wird von $600 bis zum Ur-PD gar nicht erst nach XBRA- oder XNAM-Strukturen gesucht. (Für alle Zwecke, bei denen es nur um

```
121:            bsr.s      space          ;Zwischenraum
122:            lea        12(a5),a0      ;es ist die Adr.
123:            move.l     a0,d3          ;xb_newvec
124:            bsr.s      prt_hex        ;ausgeben
125:            moveq      #-1,d7         ;flag setzen
126:            bra        next_2         ;weitersuchen
127:
128: thats_it:  lea        crlf(pc),a0    ;CR LF
129:            bsr.s      conws          ;ausgeben
130:            tst        d6             ;flag
131:            beq        loop           ;weiter geht's
132:            rts
133:
134: special:   lea        the_prg(pc),a0 ;Text laden
                                           (mit CR LF)
135:            bsr.s      conws          ;ausgeben
136:            moveq      #-1,d7         ;set flag
137:            bra        next_1         ;evtl. weiter-
138:                                      ;suchen (*)
139: task_2:    lea        slot_txt(pc),a0 ;Text laden
140:            bsr.s      conws          ;und ausgeben
141:
142:            pea        slotties(pc)   ;Subroutine
143:            move       #$26,-(sp)     ;Supexec
144:            trap       #14            ;XBIOS
145:            addq       #6,sp
146:
147: task_3:    lea        mag_txt(pc),a0 ;Text laden
148:            bsr.s      conws          ;ausgeben
149:            lea        flag_2(pc),a4  ;vorbereiten
150:
151:            pea        magic_pk(pc)   ;Subroutine
152:            move       #$26,-(sp)     ;Supexec
153:            trap       #14            ;XBIOS
154:            addq       #6,sp
155:
156:            tst.b      (a4)           ;flag testen
157:            bne.s      term           ;dann nicht
158:
159:            lea        none(pc),a0    ;'keine da' Text
160:            bsr.s      conws          ;ausgeben
161: term:      bsr.s      cnecin         ;warte auf Taste
162:            clr        -(sp)          ;Pterm0
163:            trap       #1             ;GEMDOS
164: * ------------------------------------------------
165: space:     lea        space_1(pc),a0 ;Zwischenraum
166: conws:     pea        (a0)           ;Stringadresse
167:            move       #9,-(sp)       ;Cconws
168:            trap       #1             ;GEMDOS
169:            addq       #6,sp          ;SP restaurieren
170:            rts
171: * ------------------------------------------------
172: cnecin:    move       #7,-(sp)       ;Cnecin
173:            trap       #1             ;GEMDOS
174:            addq       #2,sp
175:            rts
176: * ------------------------------------------------
177: cconout:   move       d0,-(sp)       ;char
178:            move       #2,-(sp)       ;Cconout
179:            trap       #1             ;GEMDOS
180:            addq       #4,sp
181:            rts
182: * ------------------------------------------------
183: * Langwort in d3 in Hex (als Text) auf Konsole
     ausgeben
184:
185: prt_hex:   moveq      #7,d4          ;8 Nibble
186: nibble:    rol.l      #4,d3          ;rotieren
187:            move.l     d3,d0          ;holen
188:            andi.b     #$F,d0         ;maskieren
189:            addi.b     #'0',d0        ;in ASCII wandln
190:            cmpi.b     #'9',d0        ;na
191:            ble.s      put_out        ;weg damit
192:            addq.b     #7,d0          ;sonst für A...F
193: put_out:   bsr        cconout        ;ausgeben
194:            dbf        d4,nibble      ;weiter geht's
195:            rts                       ;und zurück
196: * ------------------------------------------------
197: sorry:     lea        lorez(pc),a0   ;'sorry, min scr
                                           size 80 * 25!'
198:            bsr        conws          ;ausgeben
199:            bra        term           ;das war's
200: * ------------------------------------------------
201: * Magische Päckchen finden, ähnlich der Routine
     aus dem Betriebssystem
202:
```

→

XBRA geht und das ganze Drumherum nicht interessiert, habe ich - als 'spinoff' sozusagen - das Programm FINDXBRA erstellt.)

Nun aber genug der Theorie! Lassen wir das Programm auf uns wirken und den neuen 'XNAM-Standard' sich in Windeseile verbreiten (mir wird schon ganz weich in den Knien!). - Die hier vorgestellte Version von BASEFIND.TOS gibt übrigens nur über GEMDOS aus, daher läßt sich die Bildschirmausgabe mit einer Shell o.ä. einfach in eine Datei umleiten und bei Bedarf ausdrucken.

Viel Spaß also mit BASEFIND.TOS und erhellende Einblicke in das, was in den Tiefen des Speichers vor sich geht!

BR

Anmerkungen und Referenzen:

1. system_1.inf...system_8.inf, vom Autor exzerpierte INFO_ATARI16, Beiträge von ATARI Corp. Systemprogrammierern und anderen auf USENET, gegen Übersendung von selbstadressierter, frankierter und formatierter Leerdiskette erhältlich bei der Hamburger Microcomputer Hochschulgruppe (HMH), Postfach 630 133, 2000 Hamburg 63.

2. Jankowski,Reschke,Rabich, ATARI ST Profibuch, 1. (überarbeitete) Aufl. '88, Düsseldorf (SYBEX), 'XBRA' auf S.915, 'magic pack' auf S.179 f. (die dort vorgeschlagene Stelle ab $600 zu belegen, empfiehlt sich übrigens nicht, denn dieser Platz ist für ATARIs Betriebssystem-Patches reserviert! Ebenfalls: Vorsicht, Flexdisk!)

3. z.B. Claus Brod in diversen Scheibenkleister-Treibern, Stefan Eissing und Gereon Steffens in der Gemini-Shell, Marcel Waldvogel im AusGuck, meine Wenigkeit im Virdog und viele, viele andere... Es gibt sogar schon ein Virus (traurig!), das sich 'korrekt' über XBRA einklinkt...

4. z.B. das sehr umfassende VECSHOW.TTP von Stefan Gerle (hat übrigens einen Bug in der Line_A-Routine, was man aber nur mit RAM-TOS bemerkt), oder XBRA.TOS von Andreas Kohler, ST Computer, 1/90, S.137 ff.

5. Programme könnten sich z.B. im Wettbewerb um bevorzugte Plätze in der verketteten Liste (erster oder letzter Platz), 'umhängen', um auf diese Weise unabhängig von der Reihenfolge ihres Starts zu werden. In letzter Konsequenz würde das dazu führen, daß Programme versuchen, sich gegenseitig 'niederzumachen'.

6. Das ist wahrscheinlich auch der Grund, warum die Systemprogrammierer der ATARI Corp. nichts damit am Hut haben. In den neueren ATARI-Hilfs- und Patch-Programmen war jedenfalls von XBRA keine Spur zu finden! Das jüngste mir bekannte, POOLFIX3 von Allan Pratt (Fixprogramm für TOS 1.4 und 1.6, vom 19.1.'90), hat es allerdings (ts, ts!).

7. Kramer, Riebl, Hübner: Das TOS_Listing, Bd.1, '88, Hannover (Heise) Speicherverwaltung: S.42 ff., Programmverwaltung: S.58 ff. Der 'magic_pk' Code steht auf S.93, die Beschreibung auf S.19 ff.


```
203:  magic_pk: movea.l   $42E,a3       ;von phystop
204:  repeat:   suba      #$200,a3      ;nach unten
205:            cmpa.l    #$400,a3      ;bis $600 einsc.
206:            beq.s     finish        ;untersuchen
207:            cmpi.l    #$12123456,(a3) ;magic
208:            bne.s     repeat        ;weitersuchen
209:            cmpa.l    4(a3),a3      ;Adresse magic
210:            bne.s     repeat        ;n. Versuch
211:            moveq     #0,d0         ;löschen
212:            movea.l   a3,a1         ;zum Zählen
213:            move      #$FF,d1       ;512 Byte
214:  sum_up:   add       (a1)+,d0      ;Checksumme
215:            dbf       d1,sum_up     ;berechnen
216:            cmp       #$5678,d0     ;noch'n magic
217:            bne.s     repeat        ;war's nicht
218:            move.l    a3,d3         ;doch
219:            bsr       prt_hex       ;ausgeben
220:            bsr       space         ;Zwischenraum
221:            st        (a4)          ;flag setzen
222:            bra.s     repeat        ;und so weiter
223:  finish:   rts                     ;und zurück
224:  * ---------------------------------------------
225:  memtop: move.b  $424,d0           ;memcntrl holen
226:          andi    #$F,d0            ;low nibble
227:          lea     mem_list(pc),a0   ;memconf-Tabelle
228:          moveq   #count/6-1,d1     ;Eintrag 6 Byte
229:  search: cmp     (a0)+,d0          ;Eintrag da?
230:          beq.s   recogn            ;gefunden?
231:           addq   #4,a0             ;Wert überspringn
232:          dbf     d1,search         ;weitersuchen
233:
234:  recogn: move.l  (a0),d0           ;RAM-GröPe
235:          bne.s   skip_3            ;stattdessen
236:          move.l  $42E,d0           ;phystop nehmen
237:  skip_3: subi.l  #16,d0            ;Buserr vermeiden
238:          rts
239:  * ---------------------------------------------
240:  slotties: move      $454,d5       ;nvbls
241:            movea.l   $456,a4       ;_vblqueue
242:            tst       d5            ;nvbls
243:            beq.s     finis         ;kein Slot?retrn
244:            moveq     #0,d7         ;flag löschen
245:            subq      #1,d5         ;Zähler f. dbf
246:  getslot:  move.l    (a4)+,d3      ;Eintrag holen
247:            tst.l     d3            ;steht was drin?
248:            beq.s     next_3        ;nächstes slot
249:            bsr       space         ;Zwischenraum
250:            bsr       prt_hex       ;ausgeben
251:            moveq     #-1,d7        ;flag setzen
252:
253:            lea       xnam(pc),a2   ;XNAM-Eintrag
254:            move.l    -4(a4),a1     ;Slot-Adresse
255:            subq      #8,a1         ;Suchposition
256:            cmpm.l    (a1)+,(a2)+   ;'XNAM' da ?
257:            bne.s     next_5        ;nein -> weiter
258:
259:            lea       write(pc),a0  ;Leerstring
260:            move.l    (a1),2(a0)    ;Namen dorthin
261:            bsr       conws         ;und ausgeben
262:
263:  next_5:   tst       d7            ;flag testen
264:            beq.s     next_3        ;erstes Mal
265:
266:  next_4:   lea       slot_col(pc),a0   ;umbrechen
267:            bsr       conws
268:  next_3:   dbf       d5,getslot    ;weiter suchen
269:  finis:    rts
270:  * ---------------------------------------------
271:  mem_list: dc.w      4
272:            dc.l      $80000     ;512K
273:            dc.w      5
274:            dc.l      $100000    ;1M
275:            dc.w      6
276:            dc.l      $280000    ;2M5
277:            dc.w      8
278:            dc.l      $200000    ;2M
279:            dc.w      9
280:            dc.l      $280000    ;2M5
281:            dc.w      $A
282:            dc.l      $400000    ;4M
283:  count     =      *-mem_list    ;Tabellenlänge
284:            dc.w      0,0,0      ;Tabellenende
285:
286:  xbra:     dc.l      'XBRA',0,0
287:  write:-   dc.b      '  1234',0
288:  flag_2:   dc.b      0
```

→

8. *So etwas passiert ziemlich regelmäßig nach Verlassen gewisser Shells!*

9. *Landon Dyer: ATARI GEMDOS Reference Manual, 1986, ATARI Corp. USA*

10. *Alex Esser: TOS intern, ST Computer 1987, Sonderheft Nr.2, S.35 ff., basiert noch auf TOS 1.0. Interessant in diesem Zusammenhang auch die Serie 'Auf der Schwelle zum Licht', beginnend in der ST Computer 12/87, sowie der zweiteilige Beitrag 'Systemvariablen des TOS', ST Computer 11&12/88, des gleichen Autors.*

```
289:  empty:     dc.b      '        ',0
290:  space_1:   dc.b      '  ',0
291:  crlf:      dc.b      13,10,0
292:  none:      dc.b      'none!',13,10,0
293:  title:     dc.b      13,'basepage  prog_len
                           p_parent  XNAM  XBRA  _old_vec'
294:             dc.b      " _new_vec BASEFIND (c) MAXON
                            90",13,10,0
295:  the_prg:   dc.b      '  Obviously the program
                            BASEFIND itself!',13,10,0
296:  lorez:     dc.b      13,10,'sorry, min screen
                            size 80 * 25!',13,10,00
297:  posit:     dc.b      13,10,'
                            ',0
298:  slot_col:  dc.b      13,10,'                   ',0
299:  slot_txt:  dc.b      13,10,'active vbl_slots: ',0
300:  mag_txt:   dc.b      13,10,'magic_pk: ',0
```

```
  1:  * ---------------------------------------------------
  2:  * xbrafind.s  find XBRA structs from $600 to
        physical end of RAM (c) MAXON Computer 1990
  3:  * ---------------------------------------------------
  4:  start:     dc.w      $A000            ;line_a init
  5:             cmpi      #39,-$2C(a0)     ;v_cel_mx:
                                 minimum #columns = 40
  6:             blt       sorry            ;schade!
  7:             cmpi      #11,-$2A(a0)     ;v_cel_my:
                                   minimum #lines = 12
  8:             blt       sorry            ;traurig!
  9:
 10:             lea       title(pc),a0     ;Titelzeile
 11:             bsr       conws            ;ausgeben
 12:
 13:             clr.l     -(sp)            ;stack ->
                                             superstack
 14:             move      #$20,-(sp)       ;Super
 15:             trap      #1               ;GEMDOS
 16:             addq      #6,sp            ;restore sp
 17:             move.l    d0,-(sp)         ;ssp
 18:             move      #$20,-(sp)       ;Super ->
                                             back to user
 19:
 20:             lea       $600,a5          ;Startadresse
 21:  memtop:    move.b    $424,d0          ;memcntrl
                                             holen
 22:             andi      #$F,d0           ;low nibble
 23:             lea       mem_list(pc),a0  ;memconf-
                                             Tabelle
 24:             moveq     #count/6-1,d1    ;Eintrag ist
                                             6 Byte lang
 25:  search:    cmp       (a0)+,d0         ;Eintrag da?
 26:             beq.s     recogn           ;gefunden?
 27:              addq     #4,a0            ;Wert
                                             überspringen
 28:             dbf       d1,search        ;und
                                             weitersuchen
 29:
 30:  recogn:    move.l    (a0),d0          ;RAM-GröPe
                                             holen
 31:             bne.s     skip_3           ;stattdessen
 32:             move.l    $42E,d0          ;phystop
                                             nehmen
 33:  *           subi.l    #$8000,d0       ;ohne V-RAM
 34:  skip_3:    subi.l    #12,d0           ;Busfehler
                                             vermeiden
 35:
 36:             move.l    d0,a4            ;Endadresse
 37:
 38:  loop_1:    addq      #2,a5            ;nur gerade
                                             Adressen
 39:             cmpa.l    a5,a4            ;Endadresse
                                             erreicht?
 40:             beq       thats_it         ;fertig
 41:             move.l    xbra(pc),d0      ;Kennung
 42:             cmp.l     (a5),d0          ;steht's da?
 43:             bne       loop_1           ;weitersuchen
 44:
 45:             lea       4(a5),a0         ;xb_id
 46:             moveq     #3,d1            ;4 Byte
 47:  test_a:    cmpi.b    #' ',(a0)+       ;ASCII?
 48:             bls       loop_1           ;nein ->
                                             weitersuchen
 49:             dbf       d1,test_a        ;Langwort
 50:
 51:             move.l    8(a5),d3         ;xb_oldvec
 52:             beq.s     loop_1           ;war nichts!
 53:             tst.b     8(a5)            ;sinnvolle
                                             jmp-Adresse?
 54:             bne.s     loop_1           ;wieder
                                             nichts!
 55:             btst      #0,11(a5)        ;Adresse
                                             gerade?
 56:             bne.s     loop_1           ;total
                                             entnervt!
 57:
 58:  skip_1:    lea       write(pc),a0     ;Leerstring
 59:             move.l    4(a5),2(a0)      ;Namen
                                             dorthin
 60:             bsr.s     conws            ;und ausgeben
 61:             bsr       prt_hex          ;xb_oldvec
                                             ausgeben
 62:             bsr.s     space            ;Zwischenraum
 63:             lea       12(a5),a0        ;es ist die
                                             Adresse
 64:             move.l    a0,d3            ;xb_newvec
 65:             bsr.s     prt_hex          ;ausgeben
 66:             lea       crlf(pc),a0      ;CR LF
 67:             bsr.s     conws            ;ausgeben
 68:             bra       loop_1           ;weitersuchen
 69:
 70:  thats_it:  trap      #1               ;GEMDOS back
                                             to user
 71:             addq      #6,sp            ;restore sp
 72:             lea       end_prg(pc),a0   ;'  final
                                             address:
 73:             bsr.s     conws            ;ausgeben
 74:             move.l    a5,d3            ;Endadresse
 75:             bsr.s     prt_hex          ;ausgeben
 76:             lea       crlf_2(pc),a0    ;2*CRLF
 77:             bsr.s     conws            ;ausgeben
 78:             lea       none(pc),a0      ;Text
 79:             bsr.s     conws            ;ausgeben
 80:  term:      bsr.s     cnecin           ;warte auf
                                             Taste
 81:             clr       -(sp)            ;Pterm0
 82:             trap      #1               ;GEMDOS
 83:  * ---------------------------------------------------
 84:  space:     lea       space_1(pc),a0   ;Zwischenraum
 85:  conws:     pea       (a0)             ;Stringadr.
 86:             move      #9,-(sp)         ;Cconws
 87:             trap      #1               ;GEMDOS
 88:             addq      #6,sp            ;SP
                                             restaurieren
 89:             rts
 90:  * ---------------------------------------------------
 91:  cnecin:    move      #7,-(sp)         ;Cnecin
 92:             trap      #1               ;GEMDOS
 93:             addq      #2,sp
 94:             rts
 95:  * ---------------------------------------------------
 96:  cconout:   move      d0,-(sp)         ;char
 97:             move      #2,-(sp)         ;Cconout
 98:             trap      #1               ;GEMDOS
 99:             addq      #4,sp
100:             rts
```

→

```
101:  * ----------------------------------------------------
102:  * Langwort in d3 in Hex (als Text) auf Konsole
        ausgeben
103:
104:  prt_hex:  moveq     #7,d4          ;8 Nibble
105:  nibble:   rol.l     #4,d3          ;rotieren
106:            move.l    d3,d0          ;holen
107:            andi.b    #$F,d0         ;maskieren
108:            addi.b    #'0',d0        ;in ASCII
                                          wandeln
109:            cmpi.b    #'9',d0        ;na
110:            ble.s     put_out        ;weg damit
111:            addq.b    #7,d0          ;sonst für
                                          'A'..'F'
112:  put_out:  bsr       cconout        ;ausgeben
113:            dbf       d4,nibble      ;weiter
                                          geht's
114:            rts                      ;und zurück
115:  * ----------------------------------------------------
116:  sorry:    lea       lorez(pc),a0   ;'sorry, min
                               screen size 40 * 12!'
117:            bsr       conws          ;ausgeben
118:            bra       term           ;das war's
119:  * ----------------------------------------------------
120:  mem_list: dc.w      4
121:            dc.l      $80000    ;512K
122:            dc.w      5
```

```
123:            dc.l      $100000   ;1M
124:            dc.w      6
125:            dc.l      $280000   ;2M5
126:            dc.w      8
127:            dc.l      $200000   ;2M
128:            dc.w      5
129:            dc.l      $280000   ;2M5
130:            dc.w      $A
131:            dc.l      $400000   ;4M
132:  count     =     *-mem_list    ;Tabellenlänge
133:            dc.w      0,0,0     ;Tabellenende
134:
135:  xbra:     dc.l      'XBRA',0,0
136:  write:    dc.b      '  1234  ',0
137:  flag_2:   dc.b      0
138:  space_1:  dc.b      '  ',0
139:  crlf:     dc.b      13,10,0
140:  crlf_2:   dc.b      13,10,10,0
141:  end_prg:  dc.b      13,10,'  final address:  ',0
142:  none:     dc.b      '  no more XBRA  in memory!',
                           13,10,0
143:  title:    dc.b   13,"  xbrafind.tos    (c) MAXON
                           Computer GmbH 1990",13,10,10
144:            dc.b      '  XBRA  _old_vec  _new_vec',
                           13,10,10,0
145:  lorez:    dc.b      13,10,'sorry, min screen
                           size 40 * 12!',13,10,0
146:  * ----------------------------------------------------
```

Take 20

Hard- und Software

- 1 Geschäftsprogramme
- 2 Datenverwaltung
- 3 Finanzen & Investitionen
- 4 Tabellenkalkulation
- 5 Druckerhilfen
- 6 Textverarbeitung
- 7 Datenübertragung
- 8 Lernprogramme
- 9 Spiele & Unterhaltung
- 10 Grafik
- 11 Desktop Publishing
- 12 Musik
- 13 Heimprogramme
- 14 Sprachen
- 15 Informatik
- 16 Wissenschaft & Technik
- 17 Mathematik & Statistik
- 18 Utilities
- 19 Accessories
- 20 Verschiedene

Take 20 – Tip des Monats: (20)

AT-Speed **DM 598,– ***
So zuverlässig wie der PC-Speed Emulator und flink wie ein AT.

DMA-Kabel **DM 89,– ***
Mit automatischer Einschaltverzögerung für die Festplatte.

FolioTalk **DM 98,– ***
Erledigt die Datenübergabe vom Portfolio zum ST (inkl. Software).

DMA-Switcher **DM 498,– ***
Zum Betrieb von 2 ST an einem Laserdrucker, oder 2 ST an einer HD-Plus Festplatte.

DMA-Buffer **a. A.**
Für ruhiges Arbeiten: 6m Kabel schaffen die gewünschte Distanz zum Laser- und Plattenlüfter.

3,6m Tastaturkabel **DM 49,– ***

40W40 **DM 4198,– ***
Schnelle 40MB Festplatte mit integrierter 40MB Wechselplatte. 19ms Zugriffszeit, 64k Cache

(1)

Privatliquidation / Praxibuchhaltung
Aus der Praxis für die Praxis. **je DM 375,– ***

ReProk **DM 598,– ***

Date Access **DM 49,– ***
Universeller Terminplaner mit Adressenverwaltung , Notizblock, Suchfunktionen uvm...

NotDatMan **DM 198,– ***
Noten/Daten-Manager für Lehrer. Komfortable Notenverwaltung, Auswertung von Korrekturlisten, Absenzenlisten, Verwaltung von Schülerdateien und zahlreiche Statistikfunktionen.

(6)

READPIC **DM 150,– ***
Lernfähiges Texterkennungssystem für Cameron Handyscanner und Geniscan. Bedienung vollständig unter GEM, hohe Erkennungsrate, Speichern des erkannten Textes im ASCII-Format und anschließender Import in Textverarbeitungs- und DTP-Programme.

Signum!2 **DM 418,– ***

Script **DM 198,– ***

Scarabus **DM 95,– ***
Fonteditor für Signum! und Script.

That's Write Junior **DM 148,– ***

That's Write **DM 328,– ***

Geben Sie Ihrem Calamus® die Fonts, die er braucht!

-------- je DM 39,– * --------

Plup, Babble, YAPPIE, Skript, ROCA

-------- je DM 59,– * --------

AltBerlin bold, Peking reg, Caslo, ROMA REG, CARDPLAY

-------- je DM 79,– * --------

AKTIVA, Rund, JILLY, GEODET, Boedet, INFRA

(2)

BILDBANK **DM 139,– ***
Komfortables Informationssystem zur Verwaltung von Bilddaten im Degas, NEO, ART und IMG-Format. Bilder, Begleittexte und Bedienungshilfe in separaten Fenstern. Beliebig viele Bilder in der Datenbank (z.B. 1000 NEO-Bilder auf 30MB Festplatte). Läuft auf allen Monitoren, auch auf Großmonitor und Big-Screen.

ADRESSWRITER **DM 149,– ***
Adressenverwaltung mit eigenem Texteditor für Serienbriefverarbeitung. Adressdaten können über ACC-Funktion in andere Programme eingefügt werden. Flexible ASCII Import-/Exportfunktionen, Telefontaktzähler für In-/Auslandsgespräche und noch vieles mehr...

(8)

Lern ST plus **DM 59,– ***
Maskenorientiertes Universal-Lernprogramm für Sprachen, Geschichte und Faktenwissen aller Art. Komfortabel und leistungsfähig.

Zusatzdisketten für Lern ST plus, je DM 20,– *

Englisch	(3000 Vokabeln)
Englisch 2	(3000 weitere Vokabeln u. Wendungen)
Französisch	(4500 Vokabeln und Wendungen)
Spanisch	(4500 Vokabeln und Wendungen)
Italienisch	(ca. 4000 Vokabeln und Wendungen)
Latein	(ca. 3000 Vokabeln und Wendungen)
Geschichte	(ca. 300 wichtige Ereignisse)
Führerschein	(ca. 400 Multiple-Choice-Fragen)

(4)

BASiCHART 1.0 **DM 198,– ***
Schnelle und komfortable Tabellenkalkulation mit integrierter Präsentations-Graphik.

BASiCALC 2.0 **DM 98,– ***

LDW Powercalc **DM 249,– ***

Logistix **DM 249,– ***

VIP Professionell **DM 149,– ***

Querdruck 2 **DM 78,– ***
Druckt Texte bis 32000 Zeichen Breite. Jetzt mit neuer Oberfläche, Tutorial, neuen Schriften, Zeichensatzeditor, verbessertem Einzelblattbetrieb, als Accessory, WP-Modus aus WordPlus und Ausgabe auf ATARI-Laser.

(8)

STransPlus **DM 99,– ***
Elektronisches Wörterbuch mit 20.000 englischen Vokabeln. Per ACC-Funktion auch aus Textprogrammen bequem erreichbar. Die Integrierte Übersetzungshilfe liest ASCII-Text ein und zeigt gefundene Übersetzungen eines Wortes an. Einfache "Roh"-Übersetzungen (ohne grammatikalische, syntaktische oder inhaltliche Überprüfung) erstellt STransPlus auf Wunsch automatisch. Mit Vokabeltraining.

Die vorgestellten Produkte erhalten Sie bei einem der folgenden Anbieter:

HD Computertechnik
Pankstr. 42
1000 Berlin 65
030 / 4657028-29

Computer Technik Kieckbusch GmbH
5419 Vielbach
02626 / 78336 (Fax: 78337)

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt 90
069 / 763409

IDL Software
Lagerstraße 11
6100 Darmstadt 13
06151 / 58912

KREATIV-Software
Oberwürzbacher Str. 10
6676 Mandelbachtal
06803/3850 u. 06805/2666

(6)

STEVE 3 **DM 498,– ***
Integriertes Programm mit Textverarbeitung, Datenbank, Grafik, Deskop Publishing und CAI (Computer Aided Instructions). Einfache Serienbrieferstellung, Textbausteine, Rechenmöglichkeiten im Text, Übersetzungsfunktionen, Datenbank mit der Möglichkeit Bilder einzubinden, mischen von Text und Grafik, Formblatterstellung, Wecker, und vieles mehr.

STEVE-Extra **DM 68,–**
Lehrbuch zu STEVE, inklusive Beispieldiskette.

(8)

DR. SCHELM **DM 59,– ***
Fesselndes Quizprogramm für die ganze Familie. Update-Version: jetzt mit 2-Spielermodus (Doppelschelm), 600 Fragen aus 30 Wissensgebieten und zusätzlicher Spielvariante "Super-Hangman". Lehrreiche und humorvolle Unterhaltung.

Zusatzdiskette zu DR. SCHELM **DM 15,– ***
Enthält weitere 600 Fragen aus 30 Gebieten.

Erdkunde 3.0 **DM 69,– ***

ST-Math **DM 98,– ***

Händleranfragen erwünscht ...
Wir suchen kompetente Händlerkollegen, die sich an gemeinsamen Anzeigen in ST-Computer, ST-Magazin und CICERO beteiligen möchten. Bitte sprechen Sie mit einem der nebenstehenden Anbieter.

Autoren gesucht ...
Wir suchen attraktive Soft- und Hardwareprodukte zur Vorstellung auf diesen Seiten.

Skizzenbücher von Volkmar Hoppe entdeckt! 11

Wer mit Text- und DTP-Programmen arbeitet, braucht hochwertige und aussagekräftige Grafiken zur copyright-freien Verwendung: Der Künstler Volkmar Hoppe hat seiner Phantasie freien Lauf gelassen und unzählige reale wie irreale Assoziationen in seine Arbeit einfließen lassen. „Bildhafte Wortspiele, satirische Anmerkungen, zynische Reflexionen, Sinnvolles, aber auch Spielerisches und Witziges stellen das künstlerische Ergebnis dar, das mit geübter Feder zu Papier bzw. auf die Diskette gelangt ist. Das komplette Paket umfaßt 17 Disketten.



Brainstorm 4 by Hoppe **DM 189,-** *

120 Seiten DIN A4 auf 17 Disketten. Alle Dateien im GEM Image-Format: In Calamus, 1st Word und ähnlichen Programmen direkt verwendbar. Handbuch und Konvertierungshilfe.

CADjA V.1.2 **DM 998,-** * 10

CAD-Programm der Spitzenklasse, anwendbar für Elektronik, Maschinenbau, Architektur. Plotausgabe bis DIN A0. Symbolbibliotheken verfügbar. ASCII- Schnittstelle. Direkte Weiterverarbeitung mit "CADjA-CAM". Schnittstelle zu STEVE. Leistungsmerkmale: überragend! Druckausgabe auf Nadeldrucker, Laser, Plotter. Demo: DM 50,- *

STar Designer **DM 149,-** *
ARABESQUE **DM 278,-** *

Love & Fun 13

zeigt, welcher Partner zu Ihnen paßt: In der Liebe, beim Sport und am Arbeitsplatz. Kern der Profi-Version ist die schnelle Adressverwaltung, deren gesamter Datenbestand zur Partnersuche herangezogen werden kann. Mächtige Analysemöglichkeiten, flexible Suchkriterien, einfache Bedienung und die elegante Oberfläche zeichnen Love&Fun aus. **DM 59,-** *

Dr. Lustig **DM 29,-** *
Der Psychoanalytiker. Ein toller Partyspaß.

Calamus®	**DM 798,-** *	11
Outline Art	**DM 398,-** *	
Reprostudio ST	**DM 498,-** *	
Retouche	**DM 399,-** *	
PKS-Write	**DM 198,-** *	
Calamus Fonteditor	**DM 198,-** *	
DIDOT Fonteditor	**DM 198,-** *	
Layout-Paket für Calamus	**DM 79,-** *	

Gestaltungshilfe, Pass- und Schnittmarken in DIN Standartformaten A5, A4 und A3, hoch u. quer. Mit Anleitung (auf Wunsch in Englisch).

ST-DCL **DM 298,-** * 14

Kommandozeileninterpreter + Tools ST-DCL erlaubt es, auf einem Atari ST unter einer Kommandoumgebung zu arbeiten, die der VAX/VMS-Digital Command Language (DCL) nachempfunden ist. Auch für MS-DOS lieferbar.

ST Pascal plus **DM 249,-** *
SPC Modula 2 **DM 348,-** *
STAN **DM 69,-** *
Zur statistischen Auswertung und grafischen Darstellung von Zahlen und Messreihen.

Architektur & Design **DM 59,-** * 11

Über 400 Vektor- und Rastergrafiken (GEM-/PAC-Format) für CALAMUS. Beispiele finden Sie auf diesen Seiten.



Desk Assist 4.2 **DM 188,-** * 19

Das Multifunktions-Accessory! Terminplaner mit Alarm- und Dauerterminen, Adressen- und Telefondatei für Serienbriefe, Druckerspooler, Taschenrechner, Maßumrechnung, RAM und Diskeditor, ASCII-Tabelle und noch mehr ...

SIM Rechner-Freezer **DM 35,-** *
Speichert den aktuellen Betriebszustand des ST auf Diskette ab. "Eingefrorene" Programme können schnell und einfach reaktiviert werden.

Quick-Screen **DM 49,-** *
Beschleunigt Bildschirmaufbau u. Textausgabe.

LogiMouse Pilot **DM 99,-** * 11
Präzise und zuverlässig. Ergonomisches, der Hand angepaßtes Design für ermüdungsfreies Arbeiten selbst bei langem Einsatz.

Marconi Trackerball **DM 198,-** *
100%tiger Mausersatz mit hoher Lebensdauer.

VIDI-ST **DM 398,-** *
Echtzeit-Framegrabber für alle ST. Digitalisiert Videobilder in 16 Graustufen (20ms/Bild, Low-Res.); speichert Bildfolgen (12 Bilder/sec) ins RAM. Animation mit bis zu 25 Bildern/sec. Verarbeitet NEO-, Degas- und IMG-Format.

5.25"	Laufw., 40/80 Trks.	**DM 318,-** *	20
3.5" NEC	Laufwerk	**DM 259,-** *	

Slim-Line, anschlußfertig und durchgeschliffen.

EL 3.5"	Laufwerk	**DM 248,-** *
EX 30L	Festplatte	**DM 1298,-** *
EX 60L	Festplatte	**DM 1998,-** *
EX 40W	Wechselplatte	**DM 2698,-** *
MEGA-DRIVE 40A		**DM 1398,-** *

Einbauplatte 40MB für alle MEGA ST.

ST-Uhr 2 für 520/1040ST **DM 79,-** *
Auf Soundchip steckbar, einfacher Einbau.

EM 124 Multi-Monitor (f & s/w) **DM 598,-** *

VOMBLATT **DM 89,-** * 12
Musikdidaktisches Programm zum Erlernen des Notenlesens - auch ohne Vorkenntnisse - bis Hochschulniveau. Alle Schlüssel, Ein-/Ausgabe über MIDI möglich, Prüffunktionen und mehr...

Steinberg 'Twelve' **DM 99,-** *
12 Spur Midi Sequencer. 'Twelve' ist der kleine Bruder des schon populär gewordenen Steinberg Twenty-Four.

Cubase **DM 790,-** *
Desktop Midi Recording System

RAM-Erweiterung 2,5 MB **DM 898,-** * 20
Mit 4 MB Speicherkarte (steckbar) von Weide.

RAM-Erweiterung 4 MB **DM 1498,-** *
Wie oben, jedoch mit vollen 4 MB bestückt.

A1MB Ram-Erweiterung **DM 548,-** *
A3MB Ram-Erweiterung **DM 1148,-** *
Neu: Der bereits vorhandene Speicher Ihres ST bleibt einsatzbereit. Sie benötigen somit nur A3MB zur Aufrüstung von 1 MB auf 4 MB.

MiniRAM 1MB-Erweiterung **DM 298,-** *
4-Bit organisierte Erweiterung für 260/520 ST.

Seite 27 aus Brainstorm 4

Musikinstrumente&Computer
August-Bebel-Str. 3
6840 Lampertheim 5
06241 / 80899

Hohmann & Co.
Mönchseestraße 99
7100 Heilbronn
07131 / 60048

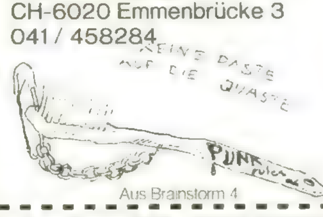
Duffner's PD-Center
Ritterstr. 6
7833 Endingen a.K.
07642 / 3875 od. 3739

T.S. Service
Szemere Hard & Software
Schleißheimerstr. 127, 8 Mü 40
089 / 3089408

Dietmar Schramm
Promberg 6
8122 Penzberg
08856 / 7287

Schick EDV-Systeme
HauptstraßeO 32a
8542 Roth
09171 / 5058-59

Bossart-Soft
Sonnenhofstr. 25 / PF 5146
CH-6020 Emmenbrücke 3
041 / 458284



Aus Brainstorm 4

* Alle Preise sind unverbindlich empfohlene Verkaufspreise

Coupon

Ich bestelle:

Name, Vorname

Straße

Plz, Ort STC 9/90

☐ Per Scheck. Zuzüglich DM 5,- Versandkosten, unabhängig von der bestellten Stückzahl.
☐ Per NN (Nur Inland, zuzügl. DM 8,- NN-Gebühr)

Textverarbeitung mit Shells



Das Betriebssystem UNIX ist bekannt für die Fülle seiner Dienstprogramme, worunter sich auch mehrere Systeme zur Textverarbeitung finden. Auf dem ST beherrschen interaktive, grafisch orientierte Textsysteme den Markt. Der Münchner Hans Kaufmann nahm sich UNIX zum Vorbild und schrieb die Programmsammlung LINGUIX, die als Public Domain und - mit erweitertem Leistungsumfang - kommerziell erhältlich ist. Für Liebhaber kommando-orientierter Shells wie Guläm bietet sich damit eine Alternative zur "konventionellen" ST-Textverarbeitung.

Im Gegensatz zu den in ST-Computer 7-8/90 vorgestellten Systemen ist LINGUIX kein eigenständiges Programm - es handelt sich um eine Utility-Sammlung, die in Zusammenarbeit mit einer Shell und Editoren ihre Wirkung entfaltet. Um Funktionen auf Texten auszuführen, muß man die jeweiligen Programme mit Optionen versehen aufrufen. LINGUIX bildet also ein kommando-orientiertes Textsystem.

Kommandeur statt Mäuschen

Hat man in einer interaktiven Textverarbeitung ein Programm laufen, dessen einzelne Funktionen man durch Befehle respektive Mausklicks anwählt, schickt man seinen Text in einer kommando-orientierten Umgebung durch mehrere Programme, die jeweils einen neuen, entsprechend verarbeiteten Text erzeugen. Durch geschickte Kombination der einzelnen Programme und entsprechende Shell-Skripts erreicht man eine Flexibilität, die eine interessante Alternative zur interaktiven Arbeit bildet. Sind die verwendeten Textfilter mit hinreichend vielen Optionen ausgestattet, kann ein kommando-orientiertes System durchaus einem Text-Editor ohne Makrosprache - wie Wordplus - überlegen sein.

Allerdings ist ein solches Textsystem völlig ungeeignet für Anwender, die nach dem What-You-See-Is-What-You-Get-Prinzip WYSIWYG - also der Darstellung des druckfähig formatierten Dokuments schon während der Texteingabe - arbeiten wollen. Man muß wissen, welches Aussehen das Dokument nach Ausführung eines Filterprogramms haben wird und natürlich auch, welche Optionen nötig sind, um die gewünschte Formatierung zu erzeugen. LINGUIX stellt also höhere Anforderungen und erfordert eine längere Einarbeitung als ein sofort zu benutzendes System wie Wordplus.

Schließlich ist natürlich zu berücksichtigen, daß in einem interaktiven Textsystem die Kommandos auf dem schon im Speicher liegenden Text arbeiten. LINGUIX arbeitet dagegen auf Dateien, die jedesmal neu gelesen und geschrieben werden müssen. Logisch, daß die Verarbeitungsgeschwindigkeit dabei entscheidend von der Geschwindigkeit des Laufwerks abhängt. Mit einem reinen Diskettensystem wird die Arbeitsleistung eher unbefriedigend sein.

Die Installation beschränkt sich auf das Hochkopieren der Utilities auf Festplatte (oder ein Arbeitslaufwerk). Sollen die LINGUIX-Programme in einem extra Ordner stehen, muß noch der Suchpfad der verwendeten Shell gesetzt werden. Beim Guläm ist dies die Variable PATH. Insgesamt werden ca. 330 kB Plattenplatz benötigt.

Mit Filter

Einfaches Anzeigen von ASCII-Dateien ist mit *head* und *cat* möglich. Zum Suchen dient ein *lgrep*, das allerdings leider nicht mit den von UNIX bekannten Regular-Expressions arbeitet. Will man die Umgebung des gefundendenen Worts sehen, kann man sich mit *kwic* und *twic* auch Zeilen vor und nach der Fundstelle anzeigen lassen.

Mehrere Filter erlauben Ersetzungen in einer Textdatei. So wandelt *tabtosp* Tabulator- in Leerzeichen um, und mit *wysilix* lassen sich Schriftattribute in einem Wordplus-Dokument verändern. Letzteres dient übrigens eigentlich der Erzeugung eines Registers: Alle Textteile mit einer bestimmten Attributkombination - also beispielsweise hell und unterstrichen



- werden mit einer Seitenangabe in eine Datei geschrieben und im Ursprungstext die Attribute gelöscht.

Mit *wpsix* kann man in ähnlicher Weise ein Register aus ASCII- oder Wordplus-Dateien vorbereiten. Dabei kommt eine zweite Datei zum Einsatz, in der die Wörter vermerkt sind, die in den Index aufgenommen werden oder - als Ausnahmeliste - dort eben nicht auftauchen sollen.

Die erzeugten Registerdateien werden anschließend mit *forminx* endgültig aufbereitet, also die Seiteneinträge für ein Stichwort zusammengefaßt. Sehr schön ist die automatische Erzeugung von "12f" aus den Seitenangaben 12 und 13 sowie "12-14", falls für das Stichwort die Seiteneinträge 12, 13 und 14 vorkamen. Und hervorragend ist die Fähigkeit von forminx, Stichwörter, die sich nur in der Endung unterscheiden, automatisch zusammenzufassen. Gibt es Einträge "Gazelle 12" und "Gazelle 14", macht diese "Flexionskontrolle" daraus eine Indexzeile "Gazelle 12,14".

Auch Inhaltsverzeichnisse lassen sich aus Wordplus-Dokumenten erzeugen. *wpinv* erkennt eine Überschrift an ihrem Beginn mit einer Zahl und Unterstreichung. In diesem Fall wird die Zeile in das Inhaltsverzeichnis mit Seitennummer übernommen. Bei der Berechnung der Seitennummern liefert die Verwendung des 1,5-fachen Zeilenabstandes übrigens falsche Werte.

Etwas ausgefallener ist der Filter *ereim*: Er gibt jeweils das letzte Wort einer Textzeile aus, auf Wunsch auch gespiegelt. Anwendungsbereich ist die Verarbeitung von Gedichten - nach einem Lauf von *ereim* hat man alle verwendeten Reimpaare gesammelt.

Zeilenorientiert sind auch von UNIX bekannte Filter wie *sort*, das eine Datei sortiert - dankenswerterweise wird die deutsprachige lexikografische Ordnung berücksichtigt (würde man nur nach Zeichencodes sortieren, kämen die Umlaute nach dem Z). *uniq* gibt die Textzeilen der Eingabedatei jeweils nur einmalig aus. Schreibt man die Ausgabe in eine Datei, hat man praktisch alle doppelten Zeilen entfernt. Mit *zdiff* kann man sich die Zeilen ausgeben lassen, die nur in Datei a auftreten, nicht aber in b. Die Ausgabe entspricht damit allen Zeilen, in denen sich a von b unterscheidet. Mit einem Schalter können auch alle Unterschiede von b zu a ausgegeben werden.

Zwischen dem ASCII- und Wordplus-Format können Dateien mit *ttow* und *wtot* umgewandelt werden. Bei der Wandlung nach ASCII kann über Schalter der Blocksatz unterdrückt und die Behandlung von Fußnoten gesteuert werden. Normalerweise werden die Fußnoten aus Wordplus am Ende der ASCII-Datei als Endnoten gesammelt ausgegeben.

Da Wordplus-Dokumente schon in ASCII gewandelt werden können, liegt es nahe, daß LINGIUX sie auch ausdrucken kann. *wprint* verwendet dabei die gewohnte Druckerinstallation in PRINTER.CFG. Praktischerweise wird diese Datei auch in den Verzeichnissen gesucht, die man in den Shell-Variablen ETC und PATH angegeben hat. Das Programm beherrscht keinen Grafikausdruck und kann leider auch nicht in eine Datei drucken.

Hat man eine einfache ASCII-Datei erstellt, kann man sie mit *clmns* zu einem Mehrspaltendokument verarbeiten. Dabei ist die Anzahl der Spalten und deren Abstand über Schalter wählbar.

reunit, das zwei sortierte Eingabedateien zu einer wiederum sortierten zusammenmischt, ermöglicht auch eine Tabellenverarbeitung. Dabei enthalten die Eingabedateien eine Text- und ein Wertespalte. Sind in beiden Eingabedateien gleichlautende Einträge in den Textspalten vorhanden, erscheint in der Ausgabedatei die Summation der beiden Wertefelder. Vielleicht sollten sich hier durch einen Schalter noch weitere arithmetische Funktionen abrufen lassen.

Das Paket umfaßt noch eine ganze Reihe weiterer Utilities, die weitere Möglichkeiten zur Textverarbeitung anbieten. Über sämtliche Programme können Sie sich in der Public Domain-Version informieren.

Da das Inout-System von TURBO-C - mit dem die Programme implementiert wurden - nicht hundertprozentig mit der TOS-Ausgabenumlenkung harmoniert, gibt es bei allen Kommandos den Schalter -o, mit dem die Ausgabe in eine Datei geschrieben wird.

Handbuch

Wie bei Produkten aus dem Shareware-Bereich üblich, gibt es auch für LINGUIX kein gedrucktes Handbuch. Die Anleitung liegt vielmehr als 133 kB großes Dokument auf der Diskette bei. Nach dem Ausdruck hat man einen fast 70seitigen Text vor sich, der erfreulicherweise mit einem dreiseitigen Register versehen ist.

Nach einer kurzen Einführung und einem Überblick der angebotenen Utilities folgt der Hauptteil der Anleitung. Hier finden sich alle Kommandos in alphabetischer Reihenfolge beschrieben. Jedem Utility ist mindestens eine Seite gewidmet, auf der in UNIX-Manier das Kommando mit Arbeitsweise, Schalterbedeutungen, meistens einem Beispiel und eventuellen Querverweisen auf andere Kommandos beschrieben wird.

Abschließend werden einige Skript-Prozeduren (Batch-Files) für Guläm beschrieben, die beispielsweise die Erzeugung eines Registers auf die Ausführung einer Kommandozeile reduzieren.

Fazit

LINGUIX ist eine interessante Sammlung kommando-orientierter Utilities, die - so man ein solches System bevorzugt - effiziente Textverarbeitung ermöglicht. Mit der Public Domain-Version hat man eine kostenlose Möglichkeit, den Funktionsumfang zu testen. Entspricht das System der individuellen Arbeitsweise, lohnt sich der geringe Aufwand für die registrierte Version. Alles in allem ein - für ST-Verhältnisse - unkonventionelles, aber funktions- und leistungsfähiges Textsystem: durchaus empfehlenswert.

RT

Bezugsadresse:

Sie finden die PD-Version von LINGUIX auf Diskette 248 der MAXON-PD-Sammlung. Die registrierte, erweiterte Version können Sie gegen DM 40,-, formatierter Diskette und frankiertem Rückumschlag beim Autor erhalten:

Hans Kaufmann
Einsteinstr. 151
8000 München 80

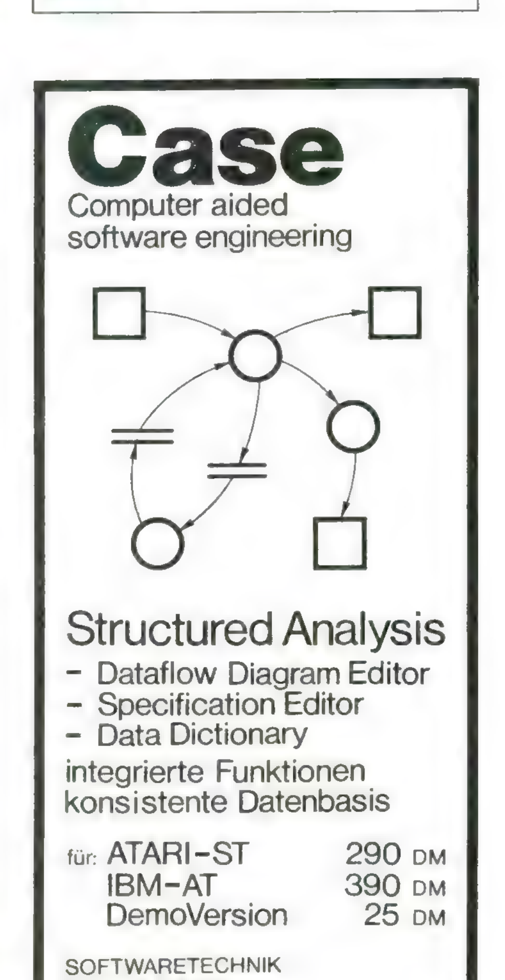

FEM + CAD + GEM® = MEANS V1®
Das FEM-System der 90er Jahre!
„MEANS ist einmalig und super, der GEM-Bedienungskomfort, die vielen Darstellungsmöglichkeiten sowie der große Leistungs- und Funktionsumfang übertreffen unsere Erwartungen bei weitem!"
Dies bestätigen uns begeisterte Ingenieure von führenden europäischen Flugzeugbauherstellern.
MEANS kann in verschiedenen Versionen für TOS- und in Kürze auch für MS-DOS-Rechner erworben werden:
Mit begrenzter Problemgröße für den Einsteiger, Privatmann und Studenten:
MEANS-I (Statik-Modul, 12 Elementtypen, 290 KB)
NETGEN (Netzgenerierungsmodul mit CAD, 465 KB)
GEOVIEW (Ergebnisauswertungsmodul, 280 KB)
begrenzt bis 50.000 Matrizenelemente: DM 225,-
oder bis 400.000 Matrizenelemente: DM 499,-
Mit unbegrenzter Problemgröße für den professionellen Bereich:
9 Module mit über 2600 KB, 34 Elementtypen, Statik, Dynamik, Temperatur, CAD-Schnittstelle, Strukturoptimierung, Strukturkopplung, 3D Raster-Hidden-Line u.v.a.
Basismodule ab DM 800,-
und mit 7 Modulen ausbaubar a. A.
Wir sind auf der Atari Messe
Bestellungen, Prospekte und weitere Informationen bei HTA-Software
Ihr Ansprechpartner: R. Schirrich, Winterstraße 10 • 1000 Berlin 51 • Telefon: 030 / 491 41 44


GESTALTEN SIE MIT CALAMUS
BRIEFBÖGEN
VISITENKARTEN
AUFKLEBER
FORMULARE
usw. ?
Mit dem neuen Gestaltungspaket von ARTWORKS COLLECTION steht Ihnen jetzt dieses und vieles mehr copyrightfrei und fertig gestaltet zur freien Verfügung!
Auf 3 Disketten, eingebunden in ca. 80 CDK-Dokumenten finden Sie:
Briefbögen
Visitenkarten
Aufkleber
Nutzeneinbindungen
jede Menge Formulare
Passermarken
Falz-u. Positionsmarken
und vieles mehr
Alles fertig gestaltet und im Handbuch mit vielen zusätzlichen Informationen zu Layout, Gestaltung und Druckvorlagenerstellung abgebildet, und das auch noch mehrfarbig!
Ab sofort lieferbar im stabilen Schuber.
Und weil so eine kleine Anzeige kein ausführliches INFO ersetzen kann, gibt's das kostenlos bei:
HESSE & HERWIG
Lesteweg 33
4790 PADERBORN
05251/92231

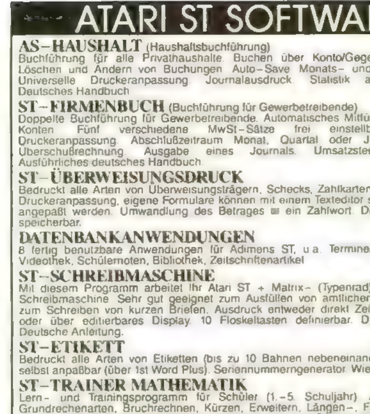

GMa-Soft
Die Komplettlösung für Ihr Büro auf ATARI ST/TT
FIBU
ST-Fibu - Finanzbuchhaltung mit Offener Postenverwaltung und freiem Kontenrahmen ab DM 348,00
Mini-Version ab DM 148,00
GMa Text - Textverarbeitung mit Mahnwesen/Serienbrief ab DM 149,00
ST Fakt - Fakturierung mit Datenübergabe an Fibu ab DM 199,00
ST Inven - Inventarverwaltung mit Datenübergabe an Fibu ab DM 79,00
Komplettpaket ab DM 699,00
kostenlose Info anfordern!
GMa Soft
Gerd Matthäus
Betriebswirt
Bergstr. 18 6050 Offenbach
Hotline von Mo. bis Sa. Tel. 069/898345


objekt-orientierte Programmierung
simula®
... back to the future!
20 Jahre Vorsprung in der Entwicklung
Aktive Objekte
Safety First
Mainframe-Qualität auf dem Schreibtisch
Apollo DN3000
Apple Macintosh/MPW
Atari ST
BESM 6 (UdSSR)
CDC 3000
CDC 6000 Cyber
Cromemco CS x00
DEC 10/20
EC 1040 1055 (DDR)
ES EVM (UdSSR)
Honeywell Bull DPS8
HP 9000/300 HP9300
IBM mainframe
IBM PC DOS
IBM PC OS2
IBM PC UNIX
IBM PC XENIX
ICL 2900
Indata IN4200
Masscomp
NCR Minitower
NCR Tower XP 32
Nixdorf Targon 31
Norsk Data NORD 100
Norsk Data NORD 500
Prime
SIEMENS 7500 BS2000
SIEMENS 7800 BS3000
SIEMENS MX
SIEMENS WX200
SIEMENS PCD
SUN 3
SUN 4 (SPARC)
Unisys 1100
VAX UNIX
VAX VMS
Mehr dazu finden Sie in c't, Ausgabe April 90, Seite 334
PC (MS-DOS) 1) 448,- (298,-)
Atari ST 1) 198,- (148,-)
GEM Desktop TOS AES und VDI Libs
deutsche Dokumentation
Apple MACII Schutzgebühr 15,-
zur Ausführung unter MPW
PC-386(UNIX) single user ca. 7700,- (4600,-)
Sun SPARCstation1 ca. 5300,- (2800,-)
Alle Simula-Systeme jetzt in Deutschland über:
simulateam GmbH iG
Ref: sc09 - Postf. 50 01 63 D-4600 Dortmund 50
...does it with class!


Speichererweiterung mit neuesten SIP-Bausteinen jetzt für jeden ST
2.5 MB 475,-
Und der Einbau?
Na gut. Machen wir den auch noch. Inklusive.
1040er, 520er - 2,5 MB 475,-
1040er, 520er - 4 MB 798,-
MEGA2 auf 4 MB 475,-
SIPs'auf 4MB 350,- 2 St.
SIMs' für ST/E 165,- p.St.
Preise inkl.Einbau * Versand +16,- (Wertpaket)
Warum SIPs' ?
* kleinste Bauform: 4 MB auf nur 8 x 5cm
* keine Platzprobleme mit anderen Erweiterungen
* einfachster Ausbau: 2 SIPs' in unsere 2.5 MB Erweiterung stecken = 4 MB (für 350,-)
* zuverlässig durch Qualitätsbausteine
Weitere Informationen im Kleinanzeigenteil dieser Ausgabe
Sven Betz
Tel. 040 / 420 43 63
2000 Hamburg 20 * Hohe Weide 50


Case
Computer aided software engineering
Structured Analysis
- Dataflow Diagram Editor
- Specification Editor
- Data Dictionary
integrierte Funktionen
konsistente Datenbasis
für: ATARI-ST 290 DM
IBM-AT 390 DM
DemoVersion 25 DM
SOFTWARETECHNIK
Dipl.-Ing. U. Böhnke
Lindenstr. 31
8012 Ottobrunn
Tel. 089/609 2449





ATARI

1040 STE / SM 124	DM	1348,–
Mega ST 1 / SM 124	DM	1348,–
Mega ST 2 / SM 124	DM	2128,–
Mega ST 4 / SM 124	DM	3098,–
Megafile 30	DM	848,–
Megafile 60	DM	1178,–
Megafile 44 incl. Cartridge	DM	1998,–
Laser SLM 804	DM	2348,–
SM 194 19" Monitor für ST's	DM	3748,–

Alle Preise incl. Mwst.. Wir führen nur deutsche Originalware direkt vom autorisierten ATARI - Händler und geben Ihnen ein volles Jahr Garantie!

KRÜGER EDV - MARKETING

Tel. 0 28 57 / 17 01 Fax. 0 28 57 / 17 00

Rees ☆ Düsseldorf ☆ Viersen

Der Public Domain Hit für Ihren ATARI

Halten Sie sich fest. Ab sofort bekommen Sie meinen thematisch geordneten PD Katalog zusammen mit 6 (sechs!) prallvollen Disks mit Super PD gefüllt als Probepaket für (20.- Schein oder V-Scheck) (Für 3.- erhalten Sie "nur" *DEN Katalog!*)

Und sonst...? erhalten Sie bei mir PD zu Bedingungen, die auch Sie überzeugen werden:

- *DER KATALOG* ist thematisch geordnet, und enthält viele nützliche Programme, die Sie endlich auch finden können (s.o.)
- PD Disketten aus den großen Serien einzeln schon für 5.00 DM, natürlich Staffelpreise
- Im Abo schon ab 3.00 DM
- Schnelle, virenfreie Lieferung
- Thematisch geordnete Spezialpakete mit der besten PD, zehn 2 DD Disks nur 35.-
- Jeden Monat die getesteten "Better Bit" Pakete für nur 30.- inkl. Versand (NN plus 4.-), siehe Anzeigen in der ST Computer

Nicht die größte PD Sammlung, aber eine der Besten ihrer Art. Überzeugen Sie sich selbst...



Andreas Mielke
EDV Software und mehr
Vinnhorster Weg 35
3000 Hannover 21
Tel. 05 11 / 75 91 56 (0-24h)

Der SteuerStar '89

Lohn- u. Einkommensteuer 89

50 ,- DM/Update 20,- DM

für alle Atari ST sw/col

Test ST-Magazin 2/89:

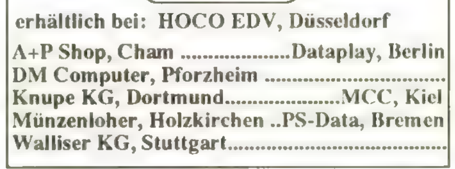
"Der SteuerStar.... nimmt ohne Zweifel einen sicheren Platz in der Reihe der Spitzensoftware für den ST ein."

Dipl. Finanzwirt J. Höfer
Grunewald 2a
5272 Wipperfürth
Tel. 02192/3368











ROMAN MODERN

Komplette und z.Z. umfangreichste Schriftfamilie für **Signum** in fünf harmonisch aufeinander abgestimmten Schriftschnitten für 24-Nadel- oder Laser-Drucker:

Roman Modern Regular
Roman Modern Bold
Roman Modern Italic
Roman Modern Bold Italic
ROMAN MODERN CAPS

Jeder Schnitt liegt in sieben Größen vor (6, 8, 10, 11, 12, 14, 16pt), verfügt über Ligaturen, Sonderzeichen und einen Grundbestand an akzentuierten Buchstaben — je Schnitt und Größe mehr als 170 Zeichen.

Als Vorlage diente die TEX-Schriftfamilie CMR, so daß nun auch Signum-Anwender eine ähnlich einzigartige typografische Ausgabequalität erzielen können.

Roman Modern komplett 130,– DM

(34 Fonts, je Font ein Standard- und ein Sonderzeichensatz, Macro's, ausführliche Dokumentation)

Gegen Verrechnungsscheck oder per Nachnahme, zzgl. 5,- DM Versandkosten bei *

Detaillierte Informationen und Schriftproben gegen 2,- DM Rück-Porto (in Briefmarken) bei *

* H.Schlicht, Ketzendorfer Weg 4H,
2104 Hamburg 92, Tel.: 040 / 7 01 64 92

QUERDRUCK2

Der Breitdrucker per Software

Beliebige Texte werden um 90° gedreht auf Matrixdruckern ausgegeben. Vor allem in Zusammenarbeit mit Datenbank-, Tabellenkalkulations- und Finanzprogrammen ergeben sich Vorteile. Wenn diese Programme ihre Druckausgabe auch auf Diskette geben können (fast immer möglich), steht einem Querausdruck von bis zu 32000 Zeichen pro Zeile nichts im Wege.

QUERDRUCK2 unterstützt alle gängigen 9- und 24-Nadeldrucker und den ATARI-Laser.

QUERDRUCK2 läuft auf allen Atari ST, auch mit den neuen TOS-Versionen, und ist nicht kopiergeschützt. Es ist im Fachhandel und durch das Entwicklungsbüro Dr. Ackermann, Kanalweg 1a, D-8048 Haimhausen, Tel. 08133/1053 erhältlich. DM 78.-, keine Versandkosten bei Vorauskasse.

Hausverwaltung

Erprobte Branchenlösung für:
Gewerbe-, Miet-, Mischobjekte
gesetzl. Nebenkostenabrechnung
Netzwerkversion verfügbar!

kleine Version	398,00 DM
bis 40.000 ME	1698,00 DM
+ Nebenkostenabrechnung	298,00 DM

IDEE Individuelle Computer-Lösungen GmbH
Waidmannstraße 12; 2000 Hamburg 50
Tel.: 040/85 50 66; Fax: 040/ 850 18 58

Fridolins RAMCARD

erweitert auch Ihren ATARI ST
auf 2 MB oder mehr!

F. Heyer
Promenadenstraße 50
D-5100 Bad Aachen
Tel.: 0241 / 35247 (24h)



Professionelle Schön-Schrift

mit **Signum** und **Laser-/24-Nadeldrucker**
→ jetzt auch für Textverarbeitung **script**

- Modernes, optimal lesbares Schriftbild
- in den Größen 8, 10, 12 und 15 Punkt,
- für Laserdrucker auch 6 und 20 Punkt,
- einschließlich unproportionaler Ziffern
- normal und fett, und dazu ein schlau
- ausgetüftelter „SONDER"-Zeichensatz

mit griechischem Alphabet (Α, α, Β, β, Γ, γ, Δ, δ, ..., Ω, ω), römischen Zahlen (I, II, III, IV, ... X, ... MCMLXXXIX), fremdsprachigen (Ä, ä, Ç, ç, Ø, ø, Œ, œ, ñ, ¿), kaufmännischen (£, ‰, ®, ©, ...) und mathematischen Zeichen (∞, ∫, ∮, ≡, ≠, ≢, <, ≤, ≥, >, ~, ≈, ≡, ±, ×, ÷, ∢ usw.) und vielem mehr (①, ... ⑨, ⑩, ... ⑲, ...↑, ↓) für die unterschiedlichsten Einsatzbereiche.

Ausführliche Info mit Schriftprobe
■ für 3,- DM in Briefmarken anzufordern bei:
Walter Schön, Berg-am-Laim-Str. 133 a, 8000 München 80, Tel. (0 89) 4 36 22 31.

Neu für Laserdrucker: senkrecht laufende 10-Punkt-Schrift

Public Domain Software ATARI ST

Wollen Sie Alles?

Oder darf es ein wenig mehr sein ?

Jetzt jeden Monat neu Das Aktuellste und Beste aus dem Public Domain Markt auf drei doppelseitigen Qualitätsdisketten für den SW-Monitor Testen Sie uns Sie werden begeistert sein ! Dazu gibt es noch den dicken Katalog der PD Soft mit Beschreibungen und Bildern von fast 800 Disketten, die Sie bei uns für nur 4,- DM das Stück (inkl. Diskette) bekommen können. Selbstverständlich führen wir auch die PD-Pool und ST Magazin Disketten

Alles zusammen für nur 10,- DM, als Schein oder V-Scheck.



Alles natürlich nur bei :



PD-Soft Rese
Beethovenstr. 8
3208 Giesen 2

Stichwort: ST 03

2,5 Megabyte

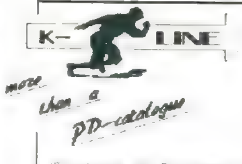
für Atari 260/520/1040ST und Mega 1.

- Bausatz mit 2-seitiger Platine (Lötstoplack)
- RAM-Fassungen mit gedrehten und vergoldeten Kontakten und Block-Kondensatoren ohne RAM's
- Komplette Einbauanleitung liegt bei.

DM 79,00

Auch für SMD- MMU's — **Nachnahme DM 7,50**

Thomas Heier
2000 Schenefeld
Gorch-Fock-Straße 33
Tel.: 040 / 830 01 99



PD-Soft

in neuen Dimensionen!

Frisch aus der Druckerei, so präsentieren wir Ihnen unseren neuen PD-Katalog K-LINE. Jedes Programm wird ausführlich beschrieben und viele Hardcopies runden unsere Erklärungen ab!

Doch das neuartige an unserem Katalog sind spannende Berichte und Tests, die ganz bestimmt Ihre Vorstellungen über einen PD-Katalog ändern werden.

Das Spielelexikon, das u. a. in dem spiralgebundenen Katalog enthalten ist, wird Sie begeistern.

Den Katalog erhalten Sie für nur DM **5,--** oder für DM **10,--** eine randvolle (2DD) PD-Disk mit dem Katalog. (Bez.: Bar, Scheck oder auch Briefmarken)

PD Ach, bei uns kostet die Public Domain Diskette (2DD) DM **4,--**

Denn PD muß PD bleiben!

COMPUTER-SERVICE KOHLER
Don-Carlos-Str. 33B, D-7000 Stuttgart 80
Tel. 07 11 / 6 78 73 92

SW-HAUSHALT V 1.0

Die private Haushaltsbuchführung für jedermann. Einfache Bedienung. Graf. Auswertungen. Monats- u. Jahresabschluß auf Drucker u. Monitor. Verwaltung von Schecks aller Art. Suchen, Ändern u. Löschen aller Buchungen. Deutsches Handbuch. Voll unter GEM nur DM 69,-

SW-VIDEO

Verwaltet 2500 Videofilme nach 24 verschiedenen Filmarten (Action, Erotik). Ermittelt alle Restlaufzeiten. Komfortables Löschen und Ändern aller Daten. Sehr schnell bei allen Such- u. Sortierfunktionen. Alle Ausgaben auf Drucker u. Monitor. Etikettendruck. Deutsches Handbuch. Voll unter GEM nur DM 59,-

Porto: VK 4,-/NN6,-

Sie erhalten bei uns über 1000 PD-Disketten. Katalog kostenlos.



SW-Software
Beethovenstr. 10
7938 Oberdischingen
Tel. 07305/8325

Ausgewählte PD-Software



Protext mit NLQ

Für alle Protext-Anwender, die neben den vielseitigen Optionen der bekannten Standardprogramme die Sortierfunktionen, Phrasendefinitionen und diversen Extras schätzen gelernt haben, entstand meist die Frage, wie man in NLQ drukken, einen eigenen Zeichensatz erstellen kann usw. Natürlich wird es über das Zusatzprogramm Prodruck ermöglicht, einen eigenen Druckertreiber zu erstellen, doch die Einarbeitungsmühe rentiert sich nur bei entsprechender intensiver Nutzung des Programms. Doch es geht auch anders:
Über die meistgenutzte Befehlstaste ESC und "G" lädt man sich den Zeichensatz ein, der in der 1. Zeile als fünftletztes Zeichen ein ESC enthält. Wird dieses ESC am Seitenanfang des aktuellen Schriftstükkes positioniert, läßt sich beispielsweise mit "x1" problemlos der Drucker auf NLQ umstellen (Star NL 10). Auch eine Kombination mit geändertem Zeilenabstand (im Programm sind ja nur 1-, $1^1/_2$- und 2zeilige Abstände vorgesehen) kann in Verbindung mit NLQ z.B. durch "ESC 2 ESC x1" vorgenommen werden. Die angegebenen Druckerbefehle sind nur eine kleine, aber wirkungsvolle Auswahl. Weitere Kombinationen bei weiteren Druckertypen unterliegen dem allseits bekannten "trial and error".

Klaus Tollkühn

Fernseher am ST

Als Ergänzung zu unserer Schaltung im ST-EXTRA 1/90 hier eine weitere Anschlußmöglichkeit. Diese hat, gegenüber der bisherigen, den Vorteil, daß sie auch mit Rechnern, die kein Composite Sync- und kein 12V-Signal haben, funktioniert. Der Nachteil ist, daß nun der Fernseher von Hand auf SCART-Empfang umgeschaltet werden muß. Die Verdrahtung ist im Bild zu sehen.

Dirk Rode

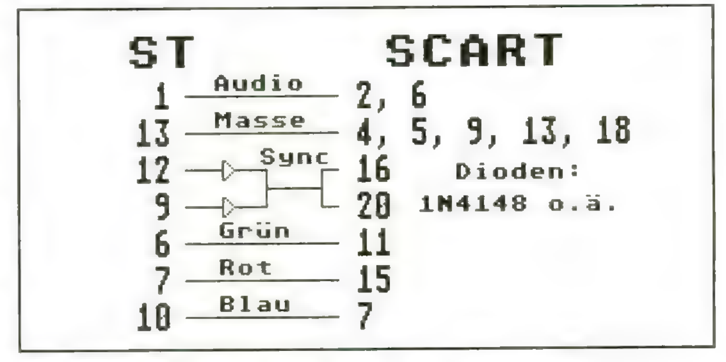


Bild 1: Verbindung ST-Monitorbuchse - SCART-Stecker

Speicherausbau beim 1040 STE

Gegenüber den übrigen ST-Modellen benötigt man beim 1040 STE keine Löterfahrung, um den internen Speicher zu verändern. Dort sind vier Speicher-Slots für SIMM-RAM-Karten vorhanden. Schwieriger wird es dann schon, wenn man vor dem Problem steht, welche SIMM-Karten man verwendet. Generell können sowohl 1MBx8- als auch 1mBx9-Karten benutzt werden. Bei letzterer liegt allerdings der gegenüber der ersten Karte zusätzliche Speicher (Datenleitung) brach. Er wird vom ST nicht erkannt. Die Zugriffszeit der RAMs sollte 130nS oder schneller sein. Werden nur zwei der vier Plätze verwendet, sind der vordere und der dritte zu bestücken. War Ihr Rechner vor dem Ausbau mit 256kB SIMM-Karten bestückt, können Sie diese leider nicht zum Ausbau auf 2,5 MB verwenden.

TW

Starten von Accessories

Viele Accessories lassen sich auch als Programm starten. Dazu muß die Endung ".ACC" durch ".PRG" ersetzt werden. Dieses Umbenennen ist aber lästig, wenn man das Accessory nur kurz ausprobieren will. Einfacher ist es, wenn man in die DESKTOP.INF-Datei der Boot-Diskette oder -Partition folgende Zeile einfügt:

```
#G 04 FF  *.ACC@ @
```

Accessories werden danach im Desktop ganz normal als Datei angezeigt, jedoch kann man sie mit einem Doppelklick starten. Accessories, die nicht als Programm gestartet werden können, führen leider nach dem Starten eine Endlosschleife aus, so daß ein Reset notwendig wird; also vorher wichtige Texte etc. abspeichern!

Thomas Mokler

Festplattenverzögerung

Wie vielen anderen Festplattenbesitzern war es mir nach kurzer Zeit zu lästig, Festplatte und Computer getrennt einzuschalten. Ich probierte die verschiedenen Lösungen fürs gemeinsame Einschalten aus, die mich nicht zufriedenstellten, da sie zu umständlich waren (Diskette im Laufwerk...). Außerdem stellte ich fest, daß meine Festplatte schneller hochfährt, wenn der Computer solange ausgeschaltet bleibt. Ich entwickelte eine kleine Schaltung, mit der der Computer verzögert eingeschaltet wird. Die Schaltung besteht aus einem Operationsverstärker LM741, einem Trimmer zur Einstellung der Zeit und einem Kondensator, welcher über einen 100 kOhm-Widerstand geladen wird. Wird der eingestellte Spannungspegel erreicht, liegt am Ausgang des OPs in etwa Betriebsspannung an. Diese Spannung macht über den Spannungsteiler den Transistor leitend, welcher dann das Relais ansteuert. Die Diode soll Induktionsspannungen kurzschließen. Das Relais muß aus Sicherheitsgründen für 220 V zugelassen sein (VDE-Zeichen) und doppelseitig abschalten. Die Sicherung arbeitet von 6 bis 24 V zuverlässig, man kann sich nach der Relaisspannung richten. Die Betriebsspannung muß nicht aufwendig stabilisiert werden, da die Zeitkonstante nicht von der Betriebsspannung abhängt. Der Trimmer läßt sich von zirka 5 bis 70 Sekunden einstellen. Wird der Trimmer auf den rechten oder linken Anschlag gestellt, wird der Transistor immer bzw. nie leitend. Die Schaltung kann in das Gehäuse des Computers oder Netzteils bei Verwendung eines 12V-Relais eingebaut werden (Vorsicht bei Schaltnetzteilen!).

Thomas Mokler

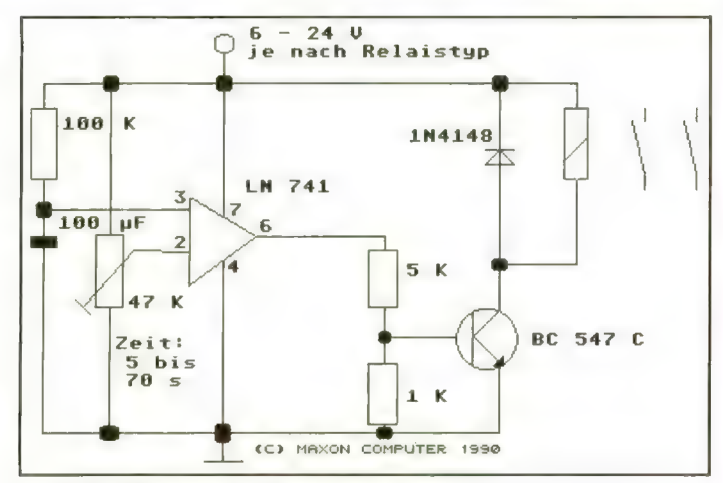


Bild 2: Der Schaltplan der Einschaltverzögerung

Bildschirmumleitung auf Drucker

Sie haben ein schönes Programm erstellt und stellen fest, daß es in gewissen Fällen angenehm wäre, den Output auszudrucken. Man kann natürlich den Bildschirminhalt mehr oder weniger bequem auf den Drucker geben. Eleganter ist jedoch die Verwendung der folgenden Aussage im Programm:

```
if (Cprnos()) Fforce(1,Fdup(3));
```

Wenn der Drucker nicht angeschlossen ist, erscheint der Output wie bisher auf dem Bildschirm. Ist er angeschlossen, wird die Bildschirmausgabe auf den Drucker umgeleitet.

Walter Fässler

Mega-Clock

Die in Mega STs eingebaute Uhr sowie baugleiche Erweiterungen für kleinere STs können mehr als nur Datum und Uhrzeit ermitteln. In diesem Chip ist zum Beispiel ein Wecker eingebaut, der leider nicht benutzt werden kann, weil ab TOS 1.2 unsinnige Werte in die Alarmregister geschrieben werden, um ein Vorhandensein der Uhr zu testen. Benutzen kann man aber immerhin die Register für Wochentag und Schaltjahr. Diese Register können nicht über XBIOS erreicht, sondern müssen direkt adressiert werden.

Wochentag:
Bit 0 in $FFFC3B löschen
(Bank 0 selektieren);
Register ist $FFFC2D

Schaltjahr:
Bit 0 in $FFFC3B setzen
(Bank 1);
Register ist $FFFC37

Das Wochentag-Register beinhaltet Werte von 0 bis 6, wobei 0 für Sonntag und 6 für Samstag steht. Das Schaltjahr-Register wird jedes Jahr um eins erhöht. Wird die Zahl drei erreicht, folgt danach eine 0, die soviel wie "Schaltjahr" bedeutet. Zu beachten ist weiterhin, daß diese Register nur 4 Bits breit sind und daß sich die CPU beim Zugriff im Supervisormodus befinden muß. Ein Omikron.BASIC-Programm soll die Programmierung verdeutlichen:

Thomas Mokler

```
 1: CLS
 2: Tag_Ermitteln
 3: PRINT "Wochentag : ";MID$("SoMoDiMiDoFrSa",
          2*Tag!+1,2)
 4: Schaltjahr_Ermitteln
 5: PRINT "Schaltjahr in ";4-Sjahr!;" Jahren"
 6: PRINT
 7: INPUT "Einstellungen richtig (J/N)";a$
 8: IF A$="n" OR A$="N" THEN
 9:   PRINT
10:   Tag_Stellen
11:   PRINT
12:   Schaltjahr_Stellen
13: ENDIF
14: END
15: '
16: DEF PROC Tag_Stellen
17:   POKE $FFFC3B,PEEK($FFFC3B) AND 14
18:   A$=CHR$(48+(PEEK($FFFC2D) AND 15))
19:   INPUT "Wochentag (0-6;0=Sonntag) :";A$
          USING "0-9-8-7x",Ret!,1
20:   IF Ret!=0 THEN
21:     POKE $FFFC2D,ASC(A$)-48
22:   ENDIF
23: RETURN
24: '
25: DEF PROC Schaltjahr_Stellen
26:   POKE $FFFC3B,PEEK($FFFC3B) OR 1
27:   A$=CHR$(48+(PEEK($FFFC37) AND 15))
28:   INPUT "Schaltjahr in .. Jahren   :";A$
          USING "+0+1+2+3x",Ret!,1
29:   IF Ret!=0 THEN
30:     POKE $FFFC37,4-(ASC(A$)-48)
31:   ENDIF
32: RETURN
33: '
34: '
35: DEF PROC Tag_Ermitteln
36:   POKE $FFFC3B,PEEK($FFFC3B) AND 14
37:   Tag!=PEEK($FFFC2D) AND 15
38: RETURN
39: '
40: DEF PROC Schaltjahr_Ermitteln
41:   POKE $FFFC3B,PEEK($FFFC3B) OR 1
42:   Sjahr!=PEEK($FFFC37) AND 15
43: RETURN
```

Listing 1: Die Ausnutzung der Mega ST-Uhr unter Omikron.BASIC

Bestimmung des Programmnamens

Es gibt Situationen, in denen man wissen muß oder möchte, mit welchem Namen das Programm gestartet wurde. Meist versucht man, im aktuellen Ordner oder in einem vorgegebenen Pfad eine entsprechende Datei zu finden. Es gibt allerdings eine wesentlich einfachere Methode ohne jegliche Dateizugriffe.

Jedes Programm besitzt eine Basepage, die direkt vor dem Programm steht und 256 Bytes beansprucht. In dieser Basepage gibt es unter anderem einen Zeiger auf die Basepage des Programms (den Parent), von dem das aktuelle Programm aufgerufen wurde, und einen Zeiger auf eine sogenannte DTA (Disk Transfer Area). In dieser DTA wird der Name des Programms vermerkt, das aufgerufen werden soll. Wenn man sich nun über die eigene Basepage den Zeiger auf die Basepage des Parents besorgt, kann man in der DTA des Parents nachsehen, mit welchem Namen das Programm aufgerufen wurde. Da man in der DTA auch die Dateilänge erfahren kann, wäre dies auch eine Möglichkeit, sich vor Linkviren zu schützen. Das funktioniert auch, wenn das Programm im AUTO-Ordner steht oder vom Desktop aufgerufen wurde, da das TOS oder Desktop auch nur Programme sind, die eine Basepage besitzen. Eine Ausnahme besteht dann, wenn das Programm nicht direkt mit Pexec() gestartet wurde, wie das die meisten Debugger machen. Jetzt zu den Zeigern:

Den Zeiger auf die eigene Basepage müssen Sie in Ihrer Dokumentation nachlesen. Der Laser C-Compiler bietet z.B. eine externe Variable "_base" an. Den Zeiger auf den Parent finden Sie ab der Position 36 ($24) in der Basepage. Der Zeiger auf den DTA befindet sich ab der Position 32 ($20). Der DTA hat folgendes Format (Offsets stehen in HEX-Zahlen am Anfang):

Axel Seeberg

```
typedef struct
    {
$00 char d res[21]:      /* reserviert */
$15 char d attrib:       /* Dateiattribut */
$16 int  d time:         /* Zeit */
$18 int  d date:         /* Datum */
$1A long d length;       /* Dateilänge */
$1E long d fname[14];    /* Dateiname */
    }
```

Der Aufbau des DTA

Startup-Code für Accessories

In einer der letzten Ausgaben hat ein Leser von Problemen berichtet, die er bei der Programmierung von Accessories mit Sozobon C hatte. Auch wenn er seine Probleme nicht genau beschrieben hat, ist die Ursache dafür mit ziemlicher Sicherheit im Startup-Code (DSTART.O) zu suchen. Für Accessories ist er in der vorliegenden Form nämlich ungeeignet, unter anderem wegen der Art und Weise, in der er die Startadresse der Basepage ermittelt. Anbei deshalb eine Minimalversion für einen Accessory-Startup-Code zusammen mit einem Testprogramm. Es ist übrigens durchaus möglich, einen Startup-Code zu schreiben, der für Programme und Accessories gleichermaßen geeignet ist. Man muß nur die Basepage auf das Vorhandensein eines Elternprozesses überprüfen. Existiert einer, wurde der Code als Programm gestartet.

Ernst-Christian Maschler

```
 1: * Desk-Accessory-Startup-Code
 2: * (c) MAXON Computer GmbH 1990
 3: *
 4: * Benutzung:
 5: * cc -c -o dastart.o dieshier.s
 6: * cc -c test.c
 7: * ld -o test.acc dastart.o test.o aesfast.a
 8:
 9:             .text
10: start:      move.l #initsp,sp  *256 Bytes v.Start
11:                                *Basepage
12:             jsr main           *main: nie zurück
13:             .bss
14:             .even
15: stack:      .ds.l   1024       *Stack=4k
16: initsp:     .ds.l   1
17:             .end
18:
19: /* Ein (nutzloses) ACC z. Test v. dastart.o */
20: #include <gemfast.h>
21: int msgbuf[8];
22: main()
23: {
24:   int ap_id,menu_id;
25:   ap_id=appl_init();
26:   menu_id=menu_register(ap_id,"  Test");
27:   while(1){
28:     evnt_mesag(msgbuf);
29:     if((msgbuf[0]==AC_OPEN) && (msgbuf[4]
                            ==menu_id))
30:       form_alert(1,"[1][Test][OK]");
31:   }/* while */
32: } /* main */
```

Listing 2: Ein Minimal-Code für Accessories unter Sozobon C

1st_Address überlistet

Die Datenbank 1st_Address ist zweifellos ein sehr gutes Produkt. Leider paßt sie die Datensätze nicht an, wenn man zusätzliche Felder einfügt oder löscht. Um nun nicht alle Datensätze mit einem Editor bearbeiten zu müssen, gibt es eine ganz einfache Lösung. Man erstellt die neue Datenbank, jedoch erst einmal unter anderem Namen. Jetzt schreibt man für die alte Datenbak eine Druckerliste für die Übertragung über den Tastaturprozessor (KEYBUF), in der alle Felder, welche die neue Datenbank enthalten soll, eingetragen sind. Hier müssen dann evtl. Felder ausgelassen oder durch ein RETURN ersetzt werden. Jetzt wird der gesamte Datensatz über "drucken" in den Tastaturpuffer umgeleitet, wobei vor dem Anklicken von OK erst die Caps-Lock-Taste gedrückt werden muß, damit der Speicher einfriert. Jetzt wird die neue Datenbank aufgerufen, F2 gedrückt und dann Caps Lock. Nun schreibt das Programm sämtliche Daten im neuen Format in die neue Datenbank. Wichtig ist aber, daß dabei kein Bildschirmschoner verwendet wird, da dieser, wenn er aktiv wird, den Vorgang solange stoppt, bis der Schirm wieder sichtbar ist (im Gegensatz zu Bildschirmuhren arbeitet 1st_Address bei einer Bildschirmabschaltung auch intern nicht weiter). Jetzt kann man die alten *.MSK- und *.DAT-Dateien löschen und die neuen umbenennen - fertig.

Ulrich Behning

Blocksatz in GFA-BASIC

Viele neue Drucker haben einen hübschen Befehl zur Blocksatzformatierung, beim Star LC 10 z.B. "ESC 32 n". Man kann zwischen jedem Zeichen einer Zeile den Abstand vergrößern und so den rechten Rand des Absatzes justieren. Der Parameter "n" soll angeben, um wieviel 1/240" der Zeichenabstand breiter wird. Leider muß man beim LC 10 ziemlich lange herumsuchen, bis man den Befehl im Griff hat. Im beistehenden Programm werden die Einschränkungen gelistet. Auch am Parameter "n" mußte nachgeschliffen werden. Die Prozedur rechnet die Länge des fehlenden Endstücks der Zeile zuerst in Zeichenbreiten, dann in "n"-Gesamtbreite aus. Da jedes Zeichen den gleichen Zuschuß bekommt, wird gerechnet, wieviel "n" jedes Zeichen haben soll und wieviel Zeichen überhaupt breiter werden sollen. Bitte beachten: N% speichert die ganzzahlige Division von Nn und L%, genauer heißt es "Nn DIV L%+1". Die Anzahl der bleibenden Zeichen wird "Anfang" genannt und bestimmt den Anfang der Zeile, die nicht geändert wird.

Jean Wantzenriether

```
 1: ' Blocksatz mit GFA-BASIC
 2: ' (c) MAXON Computer GmbH 1990
 3: '
 4: PRINT CHR$(10);CHR$(7);
 5: DATA "Der Blocksatz mit dem LC10. Achtung!"
 6: DATA "Ohne NLQ geht es nicht. Italic und"
 7: DATA "Condensed sowie Proportional sind"
 8: DATA "ausgeschlossen."
 9: DATA "PICA und ELITE"
10: DATA "in ihren drei NLQ-Styles **"
11: '
12: a%=34              !Anzahl Zeichen pro Zeile
13: mode$="PICA"       !oder ELITE
14: LPRINT TAB(10); !TAB initialisieren
15: LPRINT CHR$(27)+"x"+CHR$(1)        !NLQ ein
16: '
17: FOR i%=1 TO 6
18:   READ a$
19:   IF LEN(a$)=a%
20:     LPRINT TAB(10);a$
21:   ELSE
22:     @formatieren
23:   ENDIF
24: NEXT i%
25: '
26: PROCEDURE formatieren
27:   erst%=1          !Einrückungen bleiben
28:   WHILE MID$(a$,erst%,1)=" "
29:     INC erst%
30:   WEND
31:   lang%=LEN(a$)
32:   nn=(a%-lang%)*(20-4.28*(mode$="PICA"))
33:   n%=nn/(lang%-erst%+1)+1
34:   anfang%=lang%-nn/n%
35:   LPRINT TAB(10);LEFT$(a$,anfang%)
36:   LPRINT CHR$(27);" ";CHR$(n%);       !EIN
37:   LPRINT MID$(a$,anfang%+1,a%);
38:   LPRINT CHR$(27);" ";CHR$(0)         !AUS
39: RETURN
```

Listing 3: Blocksatz mit GFA-BASIC (ab Version 2.0) und Star L-10

TP 5.0, Olivetti und PC-Speed

Wenn Sie Turbo Pascal 5.0 unter PC-Speed V1.4 laufen lassen und die ATT Olivetti-Auflösung 640x400 monochrom benutzen wollen, ist wie folgt vorzugehen (s.Listing 4):

K. Eisenegger

```
 1: program init_graph; {Programmname}
 2: uses crt,graph;     {Unit Graph und CRT laden}
 3:
 4: var treiber, modus: integer;         {Variablen}
 5:
 6: procedure ST_hires;             {Grafikkarte n.}
 7: var treiber, modus: integer;  {m. detect abf-}
 8: begin                  {ragen! Detect installie-}
 9:   treiber:=8;          {rt EGA-Modus! Hier: ATT }
10:   modus:=5;            {640*400}
11:   initgraph(treiber,modus,'');
12: end;
13:
14: procedure detect_karte;
15: var treiber, modus: integer;  {Prozedur lief-}
16: begin                 {ert nicht max.Aufl.d.SM124}
17:   treiber:=detect;              {nämlich 640x400}
18:   initgraph(treiber,modus,'');
19: end;
20:
21: begin
22:   clrscr;             {CRT Modus Bildsch. löschen}
23:   ST_hires;           {640x400 Mono einschalten  }
24:   { hier Hauptprogramm }
25:   closegraph;                      {...schließen}
26: end.                             {...und tschüß}
```

Listing 4: Der Olivetti-Modus des PC-Speed unter TP 5.0

4 Public-Domain-Disks zum Preis von 1

Liebe(r) ST-COMPUTER-Leser(in),

wir waren selbst überrascht, so viele ATARI-ST-Besitzer(innen) mit unserem März-Spieleangebot zum Spielen verleitet zu haben. Die rege Nachfrage nach unserem über einundzwanzig Megabyte großen Spieleangebot beweist aber auch die mitlerweile meist professionelle Qualität der Public-Domain-Software. Wir fühlen uns aber auch verpflichtet, unsere Kunden von der Spielsucht wieder zu lösen und Ihnen bei allen Arbeiten, die daheim während des Spiels liegen blieb, zu helfen mit leistungsstarken Hilfsprogrammen für den Heimbereich. Dazu bieten wir musikalische Unterstützung am Computer, aber auch Druck-, Grafik- und Utility-Programme erleichtern die Arbeit am Computer erheblich. Zugegeben, ganz geheilt von der Spielsucht sind auch wir noch nicht und haben ein paar Spiele zusammengestellt und obendrein einen neuen Monochrom-Color-Monitor-Emulator dazugepackt! Alle PD-Programme sind selbstverständlich wieder nach Anwendungsgebieten in Packeten zusammengestellt. Für den Erfolg, den unser Team von sieben Mitarbeitern der PD-Abteilung durch seinen täglichen PD-Spitzenservice bei bei den PD-Bestellern erzielt hat, danken wir auch unseren Kunden.

4 gleich 1 - wie geht denn das ?

Wir versenden unsere PD-Pakete ausschließlich in kompaktierter Form. Dieses geniale Verfahren haben wir von Fsks Ludwig aufgekauft und weiter verbessert.

Jedes unserer PD-PAKETE (05....- und 06...er-Nummern) enthält vier Diskettenseiten auf je einer doppelseitigen 3.5"-Diskette. Dies wird ermöglicht durch ein in jedem Paket mitgeliefertes Entkompaktierprogramm, das in sagenhafter Geschwindigkeit (etwas über eine Min. pro 2-seitiger Disk) aus einer Diskette wieder zwei doppelseitige Disks zaubert. Die Bedienung ist kinderleicht und absolut sicher, da vollautomatisch. Schon mehr als 10.000 Kunden sind begeistert (über 100.000fach verbreitet): Außer des unschlagbaren Preises sorgt dieses System auch für Ordnung und Platzersparnis in Ihrer PD-Sammlung. Bitte beachten Sie auch Tests verschiedener Fachzeitschriften (z.B. Joystick 7/89 und 8/89).

HEIM-ANWENDUNGEN

ATARI ST sechs mal als ganz privater Computer!

Daß der ATARI ST ein perfekter Rechner sowie für den kommerziellen als auch für den wissenschaftlichen Einsatz ist, haben wir mit zahlreichen Publik-Domain- Powerpacketen bewiesen. Daß der ST auch im Spiele-Bereich ein perfekter Partner ist, hat die starke Nachfrage nach unserem letzten Einundzwanzig-Megabyte-Spieleangebot gezeigt. Was der ATARI-ST aber im Bereich der Heimanwendung als PC (Privater Computer) alles an PD-Programmen zu bieten hat, übertraf selbst unsere Erwartungen. In sechs Packeten bieten wir Ihnen alles an guter PD-Software, was unser ATARI ST auf Knopfdruck unter dem Begriff "Heimanwendung" aus seiner Datenbank zauberte. Die Vielzahl der verschiedenen Heimeinsatzbereiche ließ uns keine andere Wahl als die Sortierung der Programme nach dem Alphabet! Programme, die nicht extra gekennzeichnet sind, laufen in hoher Auflösung s/w. Für alle, die keinen Multisync- Monitor besitzen, haben wir eine besondere Überraschung: einen Monochrom - Color- Emulator der Spitzenklasse im Paket Nr. V, der sowohl einen Color- als auch einen Monochrommonitor emuliert. Wir konnten ihn einfach nicht mit allen PD-Programmen testen, aber er läuft mit einer Vielzahl von Programmen trotz der immensen zusätzlichen Rechenleistung zufriedenstellend. Nun aber ran an die Arbeit:

060060 HEIMANW. I, A-B

ADRESS-BOOK = bietet bis zu 100 Adress-Einträge incl.Telefon-Nr. und Notiz. Viele Bearbeitungsmöglichkeiten und eine Adresslabel-Druckoption machen dieses Programm interessant. ADRESS-ACC = eine Adressen-Datenbank als Accessory, immer einsatzbereit und schnell aus dem Desktop zur Hand. Ausdruck möglich. ADRESSBÜCHLE = Ein nettes elektronisches Adressbüchle incl Source in GFA-Basic 2. und 3.0. ADRESSDATEI = Adressverwaltung mit verschlüsselten Daten! Hier sind Ihre Daten gesichert! Mit Ettikettendruck. ADRESSEN = Verwaltung von Adressen mit dem Ziel, diese in verschiedenen Formen auszudrucken als Komplett-, Telefon- und Miniaturverzeichnis, letzteres in Mikroschrift. Umfangreiche Eingabe- und Bearbeitungsmöglichkeiten sowie Druckeranpassung! Mit Source Code. ALKOHOLTEST = Gestern abend getrunken und heute morgen ans Steuer? Dieses Programm berechnet den theoretischen Blutalkoholwert und die Abnahmekurve unter Berücksichtigung der persönlichen Körpermaße und informiert über Wirkung und Gefahren. Die Entscheidung und Verantwortung aber will und darf Ihnen das Programm nicht abnehmen. AIDS = ist kein verharmlosendes Spiel mit der der Gefahr. Dieses Programm wurde für das Stadtjugendamt Bad Kreuznach entwickelt. Es ist ein Informationsmedium, das im sog. "Besucherteil" ermöglicht, persönliches Wissen und Einstellung unverkrampft und informativ zu testen und gegebenenfalls zu korrigieren! Auch Leute ohne jegliche Computererfahrung können diesen Test bedienen. Neben dem "Besucherteil" besteht im eigentlichen Programm die Möglichkeit, Fragen zu ergänzen, Teststatistiken zu analysieren und Einstellungen des "Besucherteils" vorzunehmen, incl Druckeranpassung Star NL10/Epson FX 80 und kompatible. Dieses Programm ist ideal für Jugendleiter und Lehrer, aber sicher auch bei der nächsten Party am rechten Platz! ASTRO = grafisch sehr eindrucksvoll aufgemachtes Astrologie-Programm mit vielen Funktionen und Hilfen. BAR = die ideale Datenbank für den Barmixer daheim, aber auch als Stütze für den Profi (und als Ergänzung zum Alkoholtest-Prg?). Prost ST!

060061 HEIMANW. II, B-E

BAUSPAR = ein Programm, das Ihnen die Konditionen verschiedener Bausparkassen gegenüberstellt und dadurch das Angebot transparenter macht. BENZIN = Benzinkostenrechnung mit grafischer Auswertung. Gibt dadurch auch indirekt Aufschluß über Fahrweise und Motorzustand. BIO 2.6 = Biorythmusanalyse der Spitzenklasse. Sauber gestaltet, leicht bedienbar, gute Grafik und Druckmöglichkeit. BIO 3.5 = analysiert auch die Rythmenharmonie zweier Personen in den drei Bereichen Körper, Geist und Seele. BIOMAT-ST 1.0 = ein neues Rythmusanalyse-Programm, das sich gegen bestehende bewähren muß! BIORYTHMUS = ein schnelles und kurzes Programm zur Rythmusanalyse. BIORYTH-PD = Berechnung der psychisch-physisch-intellektuellen Leistungsfähigkeit, mit Source in Pascal. BUCHUS = ein kleines Buchhaltungsprogramm mit bis zu 92 Konten, in Gfa-Basic. BUDGED = ermittelt die monatliche Belastung durch regelmäpig auftretende Abzüge von Ihrem Konto. Daueraufträge werden erfapt und abgerechnet, grafische Darstellung. DATEBOOK = ein Terminkalender, der mitdenkt, automatisch bei jedem neuen Systemstart. DIA-MANAGER 1.04 = eine Datenbank zur Dia/Fotoverwaltung, sepuentiell aufgebaut, daher nur duch die Harddisk-Kapazität begrenzt. Ein eingebauter Wächter schützt vor Linkviren. An alle Möglichkeiten wurde bei der Maskenerstellung gedacht. ECONOMY = ein ganzseitiges Rechenblatt zur Berechnung von Einkünften und Ausgaben. ETAT = Planung und Kontrolle des eigenen Haushalts.Buchung, Kontenverwaltung, Monats- und Jahresplanung sowie Jahresabrechnung.

060062 HEIMANW. III, F-L

FILM = Videofilmverwaltung, eigentlich gedacht für Videotheken mit allen gängigen Möglichkeiten der Cassettenverwaltung wie Haupt- und Nebenfilm, Ein- und Austrag der Ausleiher und Listendruck. HAUSHALT = Finanzübersicht im Privathaushalt. Ein-/Ausgaben, Bilanz, Saldo. Auswertung mit Grafik. Sehr umfangreich und schön gestaltet. HEIMMANAGER = eine Kontrolle über die private Haushaltsabrechnung mit Jahresübersicht. HOME-BUDGED = verwaltet Ihre Haushaltsausgaben auf einfache Weise, indem Einkäufe und Anschaffungen gebucht und nach Warengruppen sortiert werden. Über die Monatsbilanz kann das Konsumverhalten kontrolliert werden. KALENDER = erzeugt einen Jahreskalender mit Feiertagen auf dem Bildschirm. KFZ-KZ = ermittelt je nach Eingabe Kennzeichen oder Stadt / Zulassungsstelle. KFZ-RL = eine Kfz-Kostenstatistik mit Benzinkostenrechnung. KOCH = Prüfungsfragen für angehende Köche und Hobbyköche im Multiple Choise Verfahren. Unentbehrlich auch für die perfekte Küche zuhause oder die sachlich fundierte Kritik im Restaurant. KONTO = damit Sie wissen, wo Ihr Geld bleibt! Mit Druck- und Monitorausgabemöglichkeit. KRAFTSTOFF = als kleiner Beitrag zum Umweltschutz? Verbrauchsberechnung über Benzinkosten, tabellarische und optische Auswertung, Vergleich mit dem Vormonat. Sehr gutes Konzept und Umsetzung, auch als Anregung zum umweltschonenden Spritsparen! KREUZWORT = eine Datenbank für die Umschreibungen typischer Rätselfragen. Auf Umschreibungs-Eingabe erfolgt Ausgabe möglicher Lösungen. LOTTO = nach all den Konto- und Budget-Programmen dringend notwendige Hilfe zur Finanzverbesserung! Mit diesem Programm lassen sich Lottosystemscheine auswerten nach dem Voll- und dem VEW-Kurzsystem.

060063 HEIMANW. IV, L-P

LOTTOMATIK = noch eine reelle Chance, die Hausfinanzen aufzubessern? Erstellung der Häufigkeitsverteilung bisher gezogener Zahlen. LOTTO-M = noch ein Gewinnchanchentrefferquotenerhöhungsprogramm. LOTTOBOX ST = Auswertungsprogramm für Lottosammelscheine, mit Quelltext in Omicron. LOVE 'N' FUN = wer jetzt immer noch Pech im Spiel hat, dem wird dieses Programm willkommen sein, denn es zeigt, wer in der Liebe zu Ihnen passt, aber auch in Sport, Spiel und am Arbeitsplatz. Im direkten Vergleich können bis zu vier Personen gegenübergestellt werden. Love'N'Fun macht den Grad der Harmonie in übersichtlichen Grafiken deutlich. Sie werden überrascht sein! MERKER = wo hab' ich was gelesen? Ein pfiffig gestaltetes Programm bringt Übersicht über Zeitungsartikel, Literaturhinweise und Artikel aus Fachzeitschriften. Suchen, selektieren und ausdrucken kann man nach Magazin, Jahr, Oberbegriff und vier Bemerkungen. MUSIK-MANAGER = verwaltet alle Tonträger wie LP, CD und MC nach Titel, Interpret, Aufnahmejahr und Bemerkung. Korrektur, Selektierung und Suche auf Monitor und Drucker sind selbstverständlich. PARTNERTEST = ermittelt Ihren Wunschpartner aus mehreren Möglichkeiten. Natürlich läßt sich das Programm auch als Entscheidungshilfe beim Auto-, Computerkauf usw. einsetzen, aber am schönsten ist es doch beim Ermitteln des Wunschpartners. PERSÖNLICHKEITS-TEST = dieses Programm gibt Auskunft über Ihre Persönlichkeit, Ihren Charakter und Ihre Standfestigkeit, lassen Sie sich überraschen! POSTBUSTER = Vorwahl und Gesprächsdauer eingeben und schon errechnet das Programm die Kosten, auch können Online die Gebühren am Monitor mitlaufen und der Post kann man beim Gebührenberechnen auf die Finger schauen. PRIME = wollen Sie in Ihre Zukunft sehen? Prime analysiert Ihren Namen, Geburtsdatum, Hausnummer sowie andere persönliche Daten und sagt danach Ihre Zukunft voraus. Ein netter Partyspap, den man nicht unbedingt ernst nehmen sollte.

060064 HEIMANW. V, P-S

PROFIL = legt dem Hobbypsychologen daheim Persönlichkeitsprofile von verschiedenen Personen an, die es verwaltet und vergleicht.
Profil hält sich sich genau an die Vorgaben von Zimbardo/Kirsten/Müller-Schwarz und bietet damit mehr als nur ein Hobbywerkzeug.
Komfortabel und mit Mausbedienung! READER = sind Sie es auch leid, immer alles selber lesen zu müssen? Schluß damit. Reader liest Ihnen jetzt alle (ASCII) Texte vor (am besten in englisch). RECORD = eine weitere Verwaltung von LP's, MC's und DC's, anders aufgebaut aber ebenso bequem zu bedienen. REZEPT = verwaltet und berechnet Bakkrezepte aller Art, bereits gespeicherte und Ihre eigenen nach Eingabe. Dazu stehen alle nötigen Nahrungsmittel mit ihren charakteristischen Inhaltsstoffen bereits in der Datei. RING-UP = nochmals eine Telefonkostenberechnung und -Überwachung. So bringen Sie die Telefongebühren in den Griff! SENSOR1 = testet Ihre Sinne und Ihr Gedächtnis. Farbtastenvorgaben müssen korrekt nachgespielt werden. Und weil dieser Test nur in der Farbauflösung läuft, bieten wir allen S/W-Monitorbesitzern den PD- Software-Multisync an: EMULA 5.1 = ein Emulator, der sowohl die Farb- als auch die Monochromdarstellung perfekt emuliert. Wenn auch sicher nicht mit allen, so arbeitet der Emulator doch mit viele PD-Programmen. Selbst mit vielen Spielen kommt er zurecht. Sehr überrascht hat uns auch die hervorragende S/W-Emulation am Farbmonitor. Dieses Programm, als PRG und ACC, kann nur aus Versehen oder Großzügigkeit vom Autor als PD freigegeben worden sein, man sollte es ihm in angemessener Weise danken. SPECKWEG = wurde nach den Programmen "Bar" und "Rezepte" dringend erwartet. Es ist ein freundlicher Helfer zur Beseitigung überflüssiger Pfunde. Nach ausführlicher Erfassung von Körperdaten und Abnahmewunsch wird ein Abnahmeplan erstellt und über den Kaloriengehalt vieler Nahrungsmittel informiert. Speckweg ersetztkeine Fachberatung und erwartet vom Anwender ein ausreichendes MAß an Vernunft! STAR-VIDEO 2.0 = ausgereifte, umfangreiche Videocassettenverwaltung ohne störendem Schnikschnak! Datenbearbeitung, Listendruck, Hintergrundinfos, Sortieren ...usw.

060065 HEIMANW. IV, T-Z

sTUNDENPLAN = dient zum schnellen und unkomplizierten Ausdruck eines Stundenplans. TERMINPLANER 3.2 = ein Programm, das auch als ACC läuft. Terminverwaltung mit Adressdatei, verschiedenen Kalendern und sehr umfangreichem Handbuch. TOTO = errechnet Ihnen ein Toto-System mit Gewinngarantie (so der Autor) und bedruckt die (Toto) Scheine. Sie müssen nur noch abgeben und kassieren. VIDEOARCHIV 73 = ist schon ein Oldtimer, aber immer noch perfekt in Videoverwaltung und Etikettendruck. VIP 1000 = eine Datenbank mit 1000 Namen und Geburtstagen berühmter Persönlichkeiten. Na, wer hat mit Ihnen Geburtstag? VIP wird aus der Datenbank STandart Base III gestartet, die sich auch in diesem Packet als lauffähige Demo befindet, zusammen mit der interessanten Demodatei "Autobörse" und dem ausführlichen Anleitungs-DOC. WASCHBOY = bietet allen Junggesellen endlich den richtigen Rat zur sauberen Wäsche, die dann weder verfärbt noch eingegangen oder verfilzt der Trommel entsteigt. X Y Z.

060066 DTP

Überall ist die Rede von DTP, der eigenen Hausdruckerei auf dem Computer, dem blitzschnellen Anfertigen von professioneller Paperware! Doch wer sich die gängigen Programme auf den typischen PCs ansehen mupte, der merkt bald, dap der Satz "what you see is what you get" (WYSWYG) eigentlich heipen müpte: "what you want you should never get (WYWYSNG)!" Doch gibt es Ausnahmen.Eine davon ist CALAMUS von DMC auf dem ATARI ST, einem Rechner, dessen Struktur für DTP geradezu wie geschaffen ist. Selbst den Raunheimern ist dies aufgefallen. Sie bieten den ST bereits als DTP-Komplettlösung an für Profis wie Amateure. Und tatsächlich lassen sich auf dem ATARI kinderleicht alle Wünsche erstellen, von der Visitenkarte, der Einladung, Glückwunschkarte, Flugblatt, Werbung, techn. Anleitung bis zum Buch, vom verblüffenden 24-Nadeldruck (selbst 9-Nadler werden ausgereizt) bis zum professionellen Laserbelichter von Linotype mit 2540 DPI. Und dies alles zu einem bisher unerreichten Preis/Leistungsverhältnis, eben power without the price. Dies alles sind genug Gründe für uns, Ihnen entgegen unserer ablehnenden Haltung gegenüber Demos im PD-Bereich ausnahmsweise DTP-Demoprogramme zu empfehlen, selbstverständlich voll lauffähig. Grundkenntnisse im Setzen von Schriften sind zwar von Vorteil, können aber auch mit den wachsenden persönlichen Ansprüchen schrittweise erworben werden.

CALAMUS = die neueste Demoversion eines DTP-Programms, welches auf dem ATARI Standarts gesetzt hat. Universell einsetzbar, ausbaufähig den Anforderungen entsprechend und leicht bedienbar. Umfangreicher Funktionsvorrat zur Text- und Bildbearbeitung mit Im- und Exportfunktionen und Formaten. Vektororientierte Zeichensätze, vom PD-Schriftsatz bis zum professionellen Font auf Linotype belichtet. In dieser DEMO-VERSION ist das ABSPEICHERN gesperrt. OUTLINE ART = Demo, als Ergänzung zu Calamus das künstlerische Werkzeug für den Grafiker.

060067 MIDI I

Seit die Elektronik Einzug gehalten hat bei den Musikinstrumenten, bietet sich der Computer als ideale Steuereinheit geradezu an.MIDIin und MIDIout sind die Zauberworte, nach denen Musiker Computer beurteilen. Der ATARI ST-Computer gehört zu den Rechnern, denen diese Schnittstellen von Geburt an mitgegeben wurden. Entsprechend hat er sich etabliert vom Profistudio bis hinunter zum Keyborder im Jugendzimmer. Selbst unser Tester war als Nicht-Musiker erstaunt, welche rythmisch-musikalischen Klangkonstruktionen ihm nach kurzer Einübung gelangen, mit einem einfachen Synthi und PD-Programmen. Einige der nachfolgenden Programme benötigen ein Keybord, andere lassen uns unsere Konstruktionen auch über den mageren ST-Soundchip hören und wieder andere sind reine Abspielprogramme mit Musikdemos, editiert auf kommerzieller MIDISoft- und Hardware. Auf jeden Fall aber ist der ATARI-ST mit seinen MIDI-Qualitäten zum Steuer-Herzstück vieler Bands und Partner so mancher Solisten geworden.

MSPLAYER = Musicstudio-Songplayer ist ein Programm, das Stücke über MIDI spielt, die im MS-File Format vorliegen. Alle MIDI-Kanäle sind nutzbar, umbenennen der Kanäle ist möglich (MIDI EditFunktion) incl. 91 Songs(sw und color). MIDAS = Midasmind ist ein voll funktionsfähiges Demo des Midas-Sequenzers.

Dieses Paket bekommen Sie zum Sonderpreis von NUR DM 2.--

060076 NEUE SPIELE III

Schluß mit Steineschubsen. Lapt die Kugeln rollen, entschwindet in die Welt von Abenteuer und Fantasy, oder krallt Euch die Stadt unter die Nägel.

SKOOTER = ein Arcade-Game mit anspruchsvoller Grafik und gefährlichen Farbkugeln. Aber mit etwas Geschick und Übung laßt sich immer eine Tür finden, auch wenn viele Extrahindernisse keine Ruhe aufkommen lassen. f. MORIA = fantastisches Abenteuer im Stiel von Larn oder Hack. Sie wählen sich einen Charakter und sturzen sich ins Abenteuer, nachdem Sie sich für die Reise eingedeckt haben. Doch auf der Suche nach Schätzen sind schon viele in dem geheimnissvollen Tempel verschollen. Böse Mächte, Zombies und Gnome sind eine standige Gefahr. HOTCH-POTCH = ähnlich Monopoly, allerdings mit wesentlich mehr Optionen und Ereignissen. Machen Sie Ihren Gegner bankrott, aber dazu müssen Sie ganz schön gerissen sein! f. SOKOBAN = doch noch ein Steinespiel, allerdings als einfaches Denkspiel in Labyrinthwegen gestaltet. Und natürlich EMULA 5.1 = hier aber bereits bootfahig auf der Diskette installiert!

000019 NEUE SPIELE IV

Zum Abschlup noch ein Riesenspiel, das bereits auf vielen Rechnern existiert, nur auch auf dem ATARI in der Version 3.0

Nur die Notenkapazitat ist auf 5000 Noten begrenzt. DRUMINST = ist ein Programm um ein Drumfile für den Midas-Sequenzer zu erstellen. 1STOHR = dient der musikalischen Gehörbildung. Noteneingaben und Wiederholungen von Tonfolgen lassen sich leicht vornehmen. Ein Keybord ist notwendig. DLXPIANO = dient der Soundchipprogrammierung. Die Sounds können auf einer grafisch dargestellten Tastatur abgelegt werden und sind dann mit der ST-Tastatur spielbar (color). BWSOUND = ist ebenfalls ein Sound-Editor zur Soundchip-Programmierung, incl. Beispielgeräusche.

060068 MIDI II

Nicht allein die Instrumentensteuerung hat den ATARI zum Liebling der Musiker gemacht. Wer jemals Noten gesetzt hat, und sei es nur skizzenhaft für den Klassenchor, der weip die Vorteile eines Satzprogramms zu schätzen.

SCOREPERFECT = ist ein Notensatzprogramm. Noteneingabe uber Maus, Step by Step oder MIDI-Realtime. Die vorliegende Version Level A ist eine abgespeckte Ausführung der professionellen Version von Soft Arts. Bei mindestens 1 MB Ram kann der Ausdruck in Top-Qualität sogar schon auf einem 9- oder 24-Nadeldrucker erfolgen. 5 Demo-Songs sind dabei. SYNTHI = ist zur Soundchip-Programmierung. Sequenzen lassen sich abspeichern und laden, auch als Pascal File, um in eigene Programme übernommen zu werden. FX = ein Programm für Midi-Effekte, z.B. Umdrehen der Tastatur und Transponierung. Ein Midi-Monitor ist integriert. MIDISEQ 2.0 = ist ein 1-Spur Sequencer mit Mouseplay-Funktion incl. einer FArbversion, Demos und Source Code in GFA. MUSIK PRG = ein Musik- Editor zur dreistimmigen Eingabe von Musikstücken in Notenschrift, incl. Beispielstücken. MIDIPAUKER = ist ein Lernprogramm zum Notenlesen und Anzeigen von Noten, die auf dem Keybord gespielt werden. Darstellung in Violin- oder Bap-Schlüssel. THRU = ein ACC dap als Midi-Thru Funktion softwaremäpig und in Echtzeit Daten von MIDIin an MIDIout weitergibt. RANDPLAY = komponiert 4-stimmige Stücke und sendet auf verschiedenen Midi-Kanalen. AMIDINET = dient zur Midi-Vernetzung mehrer ST's, läuft unter GEM. MIDIMIKE = ist ein eifacher MIDI-Recorder und läuft unter GEM.

060069 GRAFIK I

Nachdem nun Midi uns allen hat die Ohren abfliegen lassen zeigt der ATARI ST wozu ein Grafikrechner eigentlich gemacht wurde, bis uns die Augen herausfallen. Grafik s/w, Grafik in Farbe, Grafik klein, Grafik so groß, wie's der Speicher erlaubt, Grafik in 3-D, Grafik animiert.....

SCULPT 3D = AUTO - SCULPT 3D ist ein Editor, mit dem dreidimen-sionale Objekte fur das CAD 3D-System von ANTIC bzw: MARKT & TECHNIK erzeugt werden konnen. Der Zweck dieses Programmes ist eine Erweiterung der Möglichkeiten der Objekterzeugung für CAD-3D.Mit AUTO SCULPT 3D können nicht nur reine Spin- (Rotations-) oder Extrude- (Translations-) körper, sondern daruber hinaus Objekte durch eine Mischung dieser Generierungsroutinen erzeugt werden.Auf diese Weise kann man zu Formen gelangen, die mit den in CAD 3D enthaltenen Unterprogrammen zur Objekterzeugung garnicht, oder nur unter hohem Aufwand zu verwirklichen sind. So lassen sich mit AUTO SCULPT 3D ebenflächige Simulationen von Raum-kurven und Drehtlächen aller Art herstellen.Es können aber auch völlig unregelmapig geformte Objekte erzeugt werden.Weiter enthalt AUTO SCULPT 3D eine Routine zur Erzeugung von dreidimensionalen Sinusgraphen. Die Mit AUTO SCULPT 3D hergestellten Objektdateien können in CAD 3D geladen und dort weiterverwandt werden. SIGIPAINTER = akzeptiert sämtliche Fremdformate und malt in allen Auflösungen (sw + color). LASERPAINT = stellt neben den üblichen Zeichenfunktionen noch die Funktion "Aushöhlen" zur Verfügung, dies bietet interessante Effekte. MISTRAL = ein Zeichenprogramm mit funktionen wie: Kugelzerren, Trommelprojektion, Kreis drehen usw.

060070 GRAFIK II

Little Painter 4.28 = ist ein durchdachtes und erstaunlich vielseitiges Zeichenprogramm, von Fehlern befreit und optimiert. MEGABLIT = ein besonderes Malprogramm, die Zeichnungen können über viele Bildschirme reichen, abhängig vom Speicherplatz. Malen Sie sich eine Plakatwand! APPELHEI = nur ein Apfelmännchen-Prg.! Enorme Geschwindigkeit und die Möglichkeit riesiggrope Grafiken zu erzeugen (z.B. über 8 STAD-Screens) and volle Ausnutzung der MGE-Grafikkarte zeichnen dieses Programm aus. PCB PLATINE = Grafik in besonderer Form, als CAD. Platinenlayout mit 180 DPI Auflösung, gängigen RAstergröpen und leicht bedienbar. SUPERCAD = umfangreiches CAD-Grafikprogramm, mm-Bemapung, Symbolbibliothek, mehrere Zeichenebenen, klare Benutzeroberfläche. IFS = Grafik fur mathematisch Verspielte. IFS-Grafiken können auch natürliche Formen beschreiben wie z.B. Baumblätter.

060071GRAFIK, S/W+Color

Ohne jeglichen Hardwarezusatz ist der ATARI-ST voll farbgrafikfähig. Durch die analoge Monitoransteuerung, die jetzt auch der bekannteste PC-Hersteller mit den drei Buchstaben für seine neue Serie übernommen hat, kann man auch mit 16 (max.512) Farben begeisternde Werke in Color schaffen. Auch Videostudios und Werbeagenturen haben dies am ATARI ST erkannt und nutzen es. Daß auch gute Colorgrafik-PD-Programme immer mehr Anwender finden ist dabei nur logisch,und dap diese Programme nicht kommerziell vertrieben werden kann man oft nicht glauben.

C-CHROME = ein vielseitiges Malprogramm, das wie eine Kombination aus den Klassikern Degas und Neochrome wirkt. C-Chrome ist leicht erlernbar, kompatibel und schnell! FINELINE = feine Linien und Kurven ohne Kanten. Ecken oder Treppchen sind die Spezialität dieses Color-Programms. Degas/Neo-kompatibel und toll! PEARL = eröffnet am ATARI die Welt des Raytracing-Verfahrens. Räumliche Darstellung von Bildern und Animationen, wie man sie sonst nur von großen Rechnern kennt. Die immensen Rechenoperationen beschäftigen den ST allerdings mindestens für sechs Stunden! Das muß man sehen und erleben!

IMAGELAB = ist zwar ein S/W-Programm, konvertiert aber Low-Res-Colorbilder perfekt in die hohe Auflösung und arbeitet mit Bildgrößen von maximal 1280 * 800 Pixel bei maximal 256 Graustufen. Mit diesem Programm zur digitalen Bildbearbeitung kann der ATARI ST den großen und teuren Systemen und Programmen in den Bereichen "Image-Processing"(TM) und "Image Enhancement"(TM) durchaus Konkurrenz machen, und dies auch noch mit einem PD-Programm!

060072 GRAFIK-UTILITIES

soweit das Auge reicht, für alle und (fast) alles. Converter, Emulatoren, Server, grafikunterstützte Sprachausgaben, Schriftsätze für Grafikprogramme, Animation von Gesichtszügen, Schriften und Linien und und und...! Eine Auswahl, die wir ständig ergänzen werden!

CONVERT = ist ein nützlicher Helfer, der Bilder in fast alle auf dem ST üblichen Formate wandelt. ELITE = dient zum Einbinden verschiedener Fonts für das allseits bekannte Grafik-Programm Degas Elite. MONO.NEO = speichert den Bildschirm auf Diskette, je nach Auflösung in unterschiedlichen Formaten (Neo/Doodle). GRAFIK-SERVER = ein nützlicher Helfer für schnelle Bildwandlungen. Komprimier- und Entpack-Ruotinen und andere nützliche Arbeiten im Grafikbereich. PICTURE ARTIST 1.0 = versteht sich als Zusatzmodul der gangigsten Malprogramme und bietet neue Manipulationsmöglichkeiten fur Degas- und Neochrome-Bilder in color und s/w. FACE ANIMATOR = bringt das Gesicht einer jungen Dame in Bewegung und animiert die Gesichtszüge. Abspeichern und Filmedition sind möglich. Ein interessantes Beispiel in S/W. BOARDMOUTH = ein Gesicht liest jeden gewünschten ASCII-Text laut vor und bewegt den Mund dabei. Das Gesicht kann durch eigene digitalisierte Farbbilder ersetzt werden (lassen Sie doch mal Ihren Hund reden)! MOVIE = ist ein kleines Filmprogramm, das monochrome (s/w) Schriften und Linien in Bewegung versetzt. Und natürlich darf einer auch in diesem Paket nicht fehlen, der EMULA 5.1 = der Screen-Emulator für color und s/w nach allen Richtungen und für viele Programme, als PRG und ACC.

060073 DRUCK-UTILITIES I

Was nützt oft das schönste Programm, wenn man das Ergebnis langer Arbeit nicht zu Papier bringen kann. Wer kennt nicht die schlaflosen Nächte und ergebnislosen Diskussionen mit einem uneinsichtigen Drucker. Wer hat nicht endlich seine grafische Auswertung perfekt erstellt, nur das Textprogramm kann oder will sie einfach nicht übernehmen? Ärgerlich sind oft auch meterlange Ausdrucke mit nur einer schmalen Spalte, wo doch für mehrere nebeneinander genug Platz wäre. Oft lohnt es sich auch, für den eigenen Drucker typfremde Treiber zu laden, denn der eigene Drucker kann oft mehr als das Handbuch verrät! Und hat man endlich alles gedruckt und auf Diskette gesichert, möchte man auch gerne ein hübsches Label für sein Werk, auch daran soll's nicht fehlen!

SIGPIC.PUB = wandelt Bildschirmgrafik in ein Signum-Dokument um, welches wiederum in andere Dokumente eingefugt werden kann. LC10 COLOR = ein ganzes Packet zur optimalen Ausnutzung des Star-Colordruckers bis hin zum vierfarbigen Textdruck. Ein ACC mit ausführlicher Anleitung und ein Map für alle LC10 Color-Besitzer. NL10 SUP = ein ACC zur Einstellung von NL 10 (und LC 10) Druckern. NEC 2200 = sämtliche Einstellungen gesteuert über ein ACC ! HP = die Wordplus- und Hardcopy-Lasertreiber fur HP Laser- und Deskjet sowie Canon LPB. OKLINIT = ACC mit Steuercodes fur Oki 182, mit Source. DENSITY = das Drucker-Utility mit den verschiedensten Einstellungen. HACOMINI = druckt Degas.PI3- Bilder in 48 mm * 68 mm Gröpe. PR-EPSON = na klar, Treiber fur Epson. DRUCKER = professionelles Druckprogramm für Epson/IBM. SPC = ein SuperPosterCreator, der Screen-Bilder bis zu 3,2 * 2 m ausdrucken kann. Poster, Banner, Verzerrungen und Verformungen, alles einfach über einen 9-Nadeldrucker. Dabei werden die Pixel unter Einsatz von ASCII-Zeichen ausgedruckt. PRINTSET = beseitigt Schwierigkeiten bei der Druckeranpassung duch direkte Codeübergabe an den Drucker. PRINTER = zwei weitere allgemeine Anpassungen. F-HARDEX = !!!das Hardcopy-Prg !!! Menügesteuert, siehe auch 68000er Sonderheft Nr 19. DELTA-X = das Labeldruck- Programm überhaupt mit vielen grafischen Möglichkeiten.

060074 DRUCK-Utilities II

Sollte sich Ihr Drucker immer noch nicht ergeben haben, liefern wir hier die zweite Salve an Treibern und Programmen. Spätestens jetzt ist jeder Druckerwiderstand zwecklos.

WORDPLUS-TREIBER = je nach Wahl vom 1,5 Zeilendruck, allen Schriftarten bis zu Download-Fonts, eine Selektion der universellsten Treiber. Laden Sie doch einfach mal einen typfremden Treiber! Oft liefert dies erstaunliche Ergebnisse! PROFIPRINTER 1.2 = druckt ASCII-Dateien auf Epson-Kompatiblen sauber formatiert aus. 2-COLUMNS = druckt Texte und Listings zweispaltig aus mit verschiedenen Textattributen und Schriftgrößen. **ASL** = Programm zum Ausdruck von Textfiles und Listings. 1ST WORD-HELP = enthält eine Sammlung von Hilfsprogrammen zu 1st Word(+): Autosave zum automatischen Speichern während der Arbeit!!! BAK-KILL gegen zu viele BAK-Dateien und WPL fängt GEM-Zeichensatz ab und ersetzt ihn durch einen neuen, gut für individuelle Druckeranpassung. **PRINTER 960** = noch ein bewährter Treiber. **LQ 850** = ein komplettes Programm zur Anpassung an Wordplus incl. 1,5 Zeilendruck und voller Druckerausnutzung.

060075 NEUE SPIELE I

Eigentlich müpte nach unserem Spiele-Angebot vom März Ihr Joystick rauchen, die Maus sich am Boden krümmen und die Tastatur Ihres ST kaum noch ein müdes Klicken von sich geben. Auch wir dachten, dap alle ATARI-STs vorerst satt sein müpten. Aber böswillige PD-Programmierer geben Ihrem Rechner keine Ruhe! Schon sind wieder vier Spiele-Packete geschnürt, randvoll mit neuen Spielen, die wir in letzter Minute aus dem PD-Meer für Sie an Land gezogen haben. Und natürlich haben wir in jedes Packet den EMULAtor 5.1 gelegt für alle, die keinen Multisync haben, ausprobieren, meistens laufen die Programme trotz der immensen Zusatzarbeit für die CPU.

ADEL = die historische Simulation für 1-9 Spieler beim Kampf um die Kaiserkrone. Wirtschaft, Militar und Diplomatie sind dazu notwendig. color. ALIEN BLOCKADE = ist eine gelungene VAriante des Quixx, color. TELEBORSE = versetzt 1-4 Spieler in die Welt der Aktienspekulation.Eine spannende, realistische Simulation, s/w. SCHIEBER = ein ACC - Spiel, das jederzeit willkommene Abwechslung bietet, s/w. TEASER 1-7 = bietet neue ungewöhnliche Spielsituationen für "Full Metall Planete" von Infogrames, f. ASSAULT 1 = ein neues WK2-Scenario für das Wargame Construktion Set von SSI, wie bei Teaser wird auch hier das Originalspiel benötigt. f. BIG DEAL = ist, laut Autor, das ultimative Strategiespiel für 1-6 Spieler. Werden Sie Präsident der Galaxis, wenn Sie sich erfolgreich gegen Spione, Krieg und Sabotage wehren, aber auch Handel, Wirtschaft und Soziales sind wichtig. f. BIKER = noch eine Version des Hindernis-Ritts auf dem Motorrad. f.

090001 Signum(TM)-Zeichensätze

Inhalt: 5 Pakete (052001-052005) / Preis: KOMPLETT DM 30.-- / In diesem Mammut-Paket haben wir Ihnen über 150 auserlesene SIGNUM (TM) Public-Domain-24-Nadel-Zeichensätze zusammengestellt. Diese lassen sich mit entsprechenden Programmen auch in 9-Nadelsatze umwandeln (veröffentl. im ST-Computer 6/89). Dazu erhalten Sie noch eine Diskette mit vielen nützlichen Utilities rund um Signum (TM). Ein Paket das seinesgleichen sucht. In Preis und Leistung!!!

090002 ARTWARE BILDER

Inhalt: 9 Pakete (052006-052014) / Preis KOMPLETT DM 54.-- / Dieses Mamut-Paket macht seinem Namen wirklich alle Ehre. Über 9.000 feinster Clip-Art-Bilder im DEGAS-Format stellen wir Ihnen mit dieser Sammlung zur Verfügung. Eine fast unerschöpfliche Quelle für jeden, der sich mit Grafik beschäftigt. Ideal auch für DTP-und SIGNUM(TM) Anwender. Hier müssen Sie zugreifen. Diese Gelegenheit ist wirklich einmalig!!!

PREISE / Lieferbedingungen

Jedes Paket (05..../06....) kostet

NUR DM 8.--

Porto & Verpackung

Scheck / Bankeinzug DM 4.50

Per Nachnahme DM 6.--

Wir verwenden ausschließlich GELBE Qualitats-Disketten mit ausführlichen 4-farbigen Diskettenaufklebern. Der Mindestbestellwert beträgt DM 30.-. Fur Bestellungen unter DM 30.- berechnen wir einen MINDERMENGENZUSCHLAG von DM 6.-. Für die Diskttenbeschreibung, Inhalt und Lauffähigkeit der Programme kann trotz ausführlicher Tests und mehrfacher Überprüfung keine Gewähr übernommen werden

Händleranfragen erwünscht!!

(Händler Vertrieb erst ab Juli geplant)

pearl agency

Vermittlungsgesellschaft mbH

Hauptstrasse 29 * D-7811 Sulzburg

Tel.: 07634/69021 * Fax: 07634/6867

Optionen?

Bitte das Spiel zu machen!

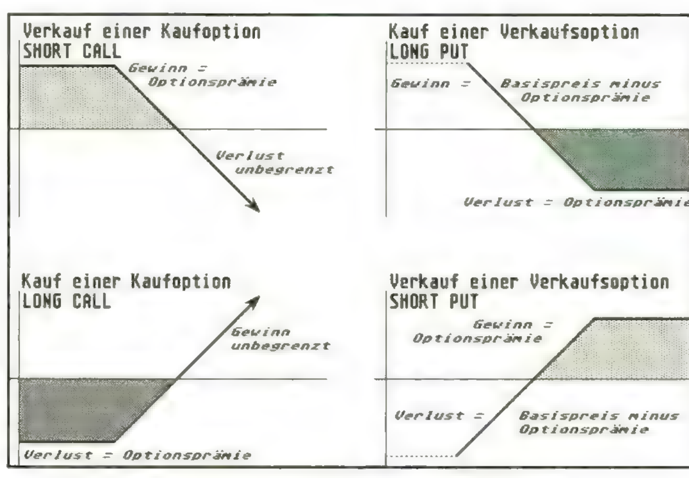


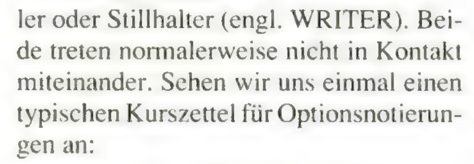
Bild 1: Die vier Grundstrategien und ihre Erfolgsaussichten

Der 26. Januar 1990 wird ein historisches Datum werden, in der Adresse Börsenplatz Nr. 6, Frankfurt am Main. An diesem Tag ging der Wertpapierhandel in Deutschland in eine neue Ära: Es gibt keinen Börsensaal mehr, Makler müssen sich nicht mehr von Angesicht zu Angesicht Orders zurufen. Dies gilt allerdings nur für eine Abart der guten alten Aktie, die Option. Gehandelt wird über Computerterminals, weshalb diese neue Options- oder Terminbörse auch gerne "Computerbörse" genannt wird. So überaus neu ist diese Einrichtung für Deutschland gar nicht. Schon vor dem ersten Weltkrieg galt der Börsenplatz Berlin als weltweit führend für Aktientermingeschäfte. Heute gilt die Börse Chicago mit täglich 500.000 Abschlüssen als Nummer eins.

Was sind denn Optionen?

Eine Option, auch Kontrakt genannt, ist eine Vereinbarung, bis zu einem bestimmten Termin eine gewisse Menge an Aktien kaufen oder verkaufen zu wollen. Hauptgedanke eines Kontraktes ist es nun aber nicht, diese Aktien auch wirklich zu liefern oder zu beziehen. Der Optionsinhaber erwartet vielmehr, durch bestimmte Marktentwicklungen und seinen anzuwendenden Strategien die Entwicklung der Option so ausnutzen zu können, daß innerhalb einer festgeschriebenen Zeitspanne Erträge erhöht oder (was leider viel häufiger vorkommt) Verluste reduziert werden können.

Es gibt zwei grundlegende Typen von Kontrakten: einerseits die Kauf- (engl. PUTs) und andererseits die Verkauf-Option (engl. CALLs). Der Käufer einer Option heißt Halter (engl. HOLDER), den Verkäufer nennt man Aussteller oder Stillhalter (engl. WRITER). Beide treten normalerweise nicht in Kontakt miteinander. Sehen wir uns einmal einen typischen Kurszettel für Optionsnotierungen an:


ALLIANZ (put) 8-2800/120; 8-2900/51; 8-3000/15; 10-2900/143; 10-3000/98; 10-3100; 12-2800/261; 12-2900/207; 12-3000/155;

Die erste Zahl gibt den Verfalltermin an, die zweite Zahl zeigt den Basispreis der Aktie, und die dritte Zahl nennt den Schlußpreis der Option. Nun müssen wir erst einmal diese vielen Fachwörter klären.

Der Verfalltermin

Optionen haben eine fest vorgeschriebene Zeitspanne, in der sie "gültig" sind. Bis zum Verfall darf der Halter (Käufer) einer Option diese "ausüben", d.h. er kann die "gezeichnete" (bestellte) Aktienmenge auch wirklich kaufen (PULL) oder muß die entsprechende Menge liefern (CALL). Letzter Termin zur "Ausübung" der Option ist immer der dritte Freitag im angegebenen Monat. Im obigen Beispiel bedeutet die "8" den Monat August. Nach dem letztmöglichen Ausübungstag ist die Option wertlos, falls der Inhaber sie nicht "glattstellt", also ein entsprechendes Anschlußgeschäft tätigt.

Der Basispreis (engl. STRIKE)

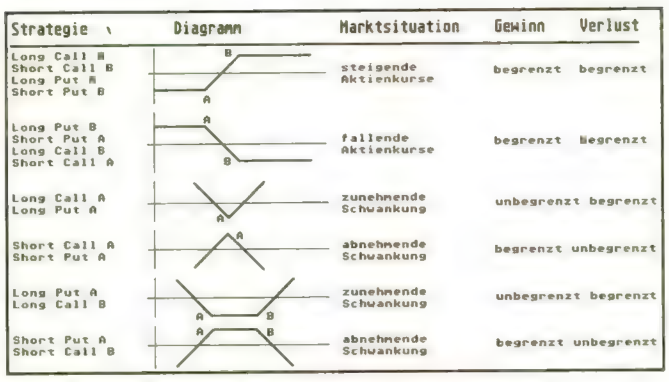
Diese Zahl ist als Festpreis zu verstehen, zu dem das Aktienpaket seinen Besitzer wechseln muß, wenn die Option ausgeübt wird. Standardmäßig ist die Zahl der Aktien auf 50 Stück pro Option festgelegt (sofern nichts anderes angegeben wird). Im Gegensatz zu den Kursschwankungen der wirklichen Aktie steht hier der Aktienpreis also von vorneherein fest.

Der Schlußpreis (engl. PREMIUM)

Das ist der Handelswert der Option, auch Optionsprämie genannt. Der Käufer (Holder) muß diese Prämie unmittelbar an den Verkäufer (Writer) entrichten. Der Schlußpreis wird sich immer mehr verringern, je näher der Verfalltermin kommt.

Das typische an diesem Optionspreis sind die starken Ausschläge in der Schwankungsbreite, auch Hebelwirkung genannt. Wenn der wirkliche Kurs der Aktie über den Basispreis z.B. bei der Kaufoption hinausgeht, sind natürlich mehr Anleger daran interessiert, diese günstigere Option zu erhalten. Deren Preis wird überproportional ansteigen. Genauso würde dies passieren, wenn der Aktienkurs tiefer fiele, als der Basispreis einer Verkaufsoption dies angibt.

Gleichgültig was geschieht, das Risiko des Optionskäufers würde aber immer nur auf die Optionsprämie beschränkt bleiben.



Strategie	Diagramm	Marktsituation	Gewinn	Verlust
Long Call A Short Call B Long Put A Short Put B		steigende Aktienkurse	begrenzt	begrenzt
Long Put B Short Put A Long Call B Short Call A		fallende Aktienkurse	begrenzt	begrenzt
Long Call A Long Put A		zunehmende Schwankung	unbegrenzt	begrenzt
Short Call A Short Put A		abnehmende Schwankung	begrenzt	unbegrenzt
Long Put A Long Call B		zunehmende Schwankung	unbegrenzt	begrenzt
Short Put A Short Call B		abnehmende Schwankung	begrenzt	unbegrenzt

Bild 2: Verschiedene Mischstrategien bei Betrachtung von Markt und Risikobereitschaft

Das Aufgeld

Zur Bewertung von Optionsscheinen werden hauptsächlich das sogenannte Aufgeld und der Hebel herangezogen. Das Aufgeld zeigt, um wieviel Prozent der Aktienerwerb über den Umweg der Ausübung einer Option teuerer oder billiger wäre, als wenn man gleich die Aktie kaufen würde. Wird z.B. das Aufgeld mit plus 10% errechnet, wäre ein Erwerb von Aktien über die Optionsausübung um 10% teurer. Man darf eben nie vergessen, der Optionsschein kostet zusätzliches Geld.

Der Hebel

Dieses Bewertungskriterium ist noch etwas komplizierter. Der Hebel gibt an, um welchen Faktor multipliziert der Optionsschein steigt oder fällt gegenüber Kurssteigerungen oder -rückschlägen der wirklichen Aktie. Ein Hebel von 3 beispielsweise sagt aus, daß wenn die Aktie um 10% schwankte, der Optionsschein um 3 * 10% = 30% schwanken würde. Das Aufgeld spielt hier eine wichtige Rolle. Je höher das Aufgeld ist, desto niedriger wird der Hebelfaktor, aber je höher der Hebel wird, desto spekulativer ist die Option.

Auf Nummer Sicher: Margins

Wenngleich dem Optionskunden in Deutschland (derzeit noch) sehr viele Hürden vor dem Anlageerfolg im Wege stehen, z.B. die Mindestdepotgröße von 20.000 DM oder die Beschränkung auf nur 14 optierbare Spitzenaktien, haben auch die Banken einige Fallstricke ersonnen: die Sicherheiten.

Allzu risikoreich ist das Spiel mit der Option, so daß die Depotbank außer der normalen Einlage (welche zum Bezahlen der Optionsprämie dient) auch noch eine zusätzliche Sicherheitsleistung in DM auf dem Konto wünscht: die Margins. Dies ist eine Art Versicherung der Depotbank gegen Verluste aus dem Optionskurs. Es ist nicht ungewöhnlich, daß die Sicherheitsleistung mindestens so hoch sein muß wie der Gesamtwert des Depots.

Der Zusammenhang Option zur Aktie

Wie unschwer zu erkennen ist, bleibt das Schicksal einer Option immer fest mit dem Kurs der wirklichen Aktie verbunden. So erscheint es nur logisch, z. B. mit einer Kaufoption und einem niedrigen Basispreis auf Kurssteigerung der Aktie zu spekulieren. Wenn nun der Aktienkurs unter dem Basispreis liegt, ist die Kaufoption "aus dem Geld". Sie ist "im Geld" (und damit günstiger), wenn die Aktie über dem Basispreis notiert. Bei den Verkaufsoptionen ist es entsprechend umgekehrt.

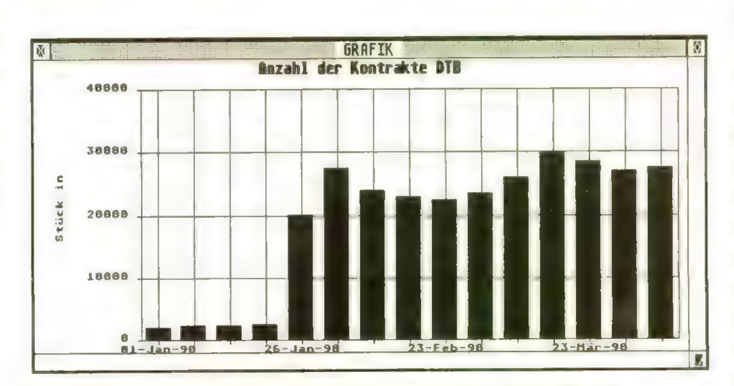


Bild 3: So stark entwickelte sich der Umsatz seit der Freigabe der Deutschen Terminbörse (DTB) am 26. Januar 1990.

Der sogenannte "innere Wert"

Der innere Wert ist jener Betrag einer im-Geld liegenden Option, um welcher der Aktienkurs den Basispreis übersteigt. Liegt die Option aus-dem-Geld, ist der innere Wert (vergleichbar mit einem Optionsgewinn) gleich Null bzw. negativ. Die Gewinnzone einer Kaufoption ist aber erst dann erreicht, wenn der Aktienkurs die Summe von Basispreis und Optionspreis übersteigt.

Der Zeitwert

Dann gibt es noch die Bezeichnung "Zeitwert". Diese Meßzahl gibt an, um wieviel die Optionsprämie den inneren Wert übersteigt, wohlgemerkt nur, wenn die Option im-Geld ist. Viel wichtiger für die Beurteilung einer Option ist der Zeitwertverfall. Am Tage des Verfalls bestimmen nur noch der Aktienkurs und der Basispreis den Wert einer Option (innerer Wert). Es ist leider typisch, daß der Zeitwert immer stärker fällt, je näher der Verfalltermin kommt.

Welche Strategie sollte man wählen?

Ein Antwort auf diese Frage kann Ihnen niemand geben. Es gibt zwar vier Grundstrategien, die für die verschiedenen Anlegermentalitäten noch abgewandelt und verfeinert werden, aber es sind viele Einzelfaktoren zu beachten.

Nach den Grundstrategien kann man ablesen, daß die Ausgangssituation für den Käufer einer Kaufoption am günstigsten erscheint. Sein Verlustrisiko ist auf den Optionspreis begrenzt und die Gewinnmöglichkeit nach oben offen. Im Gegenzug erscheint die Lage eines Optionsverkäufers insgesamt nicht sehr aussichtsreich. Seltsamerweise lehrt uns die Praxis das Gegenteil, und dafür gibt es plausible Gründe: Optionskäufer erzielen zeitweilig durchaus hohe Gewinne, dafür aber sehr selten. Verkäufer haben zwar nie "das große Los", es kommen aber regelmäßiger kleinere Gewinne herein. Käufer setzen zu sehr auf außergewöhnliche Börsentrends, die halt nicht zu häufig eintreten. Die Käufer haben zudem das Problem, daß der Wert einer Option zum Verfall hin immer an Wert verliert. Etwa ein Drittel aller Optionen verfällt wertlos. Erhebungen in den USA haben gezeigt, daß seit Einführung des börsenmäßigen Optionshandels vor 16 Jahren immer die Verkäufer das Geld gemacht haben.

Das Handeln von Optionen gleicht einem Lotteriespiel.

DK

CATO

Die Strategie wird mit uns sein

Für viele Börsianer ist es in höchstem Maße widersinnig, Entscheidungen über Geldanlage einem undurchsichtigen Elektronengehirn zu überlassen. Besonders als bekannt wurde, daß der Börseneinbruch (CRASH) im Oktober 1989 von automatisch verkaufender Börsen-Software ausgelöst wurde, scheint das Mißtrauen gegenüber zuviel Elektronik im Börsensaal weiter zuzunehmen.

So verlassen sich alte Parketthasen viel zu gerne auf den berühmten Riecher, und nur jüngere Semester greifen dann mal zu elektronischer Hilfe. Ist diese Angst vor dem alles beherrschenden Computer im Geldgeschäft wirklich angebracht? Wäre es nicht besser, sich die RAMs in unseren Zentraleinheiten so dienstbar zu machen, daß der Mensch noch immer den Überblick und damit die Macht (auch über sein Geld) behält?

Überproportional stark ist der Zulauf der Unter-Dreißigjährigen als Spekulanten in den vergangenen Jahren gewesen. Es scheint sich der Trend zu bestätigen, daß die alte Garde der Geldanleger, welche sehr stark auf Erfahrung und Wissen bauen konnte, unverhältnismäßig schnell von den sogenannten "Yuppies" (Neulinge) abgelöst wird, welche noch nicht auf ein großes Know-How zurückgreifen können. Generationsbedingt bedienen sich diese Newcomer in stärkerem Maße elektronischer Hilfe. Wenn es nun aber Computer-Programme gibt, die uns helfen, aus der Fülle der Böseninformationen die wichtigsten Faktoren herauszulesen, und der Mensch bleibt nach wie vor bestimmende Größe, dann müßte doch der Anlageerfolg perfekt sein.

Wie wir in einem früheren Test zu dem Thema Chart-Analyse in dieser Zeitschrift sehen konnten, kann es durchaus interessante und erfolgversprechende Börsen-Software geben, bei der nicht alles in einer Art "Geheimsitzung" im Hintergrund abläuft und irgend eine nebulöse Anweisung dabei herauskommt. Diese Chart-Programme waren nichts anderes als umfangreiche Statistiker mit einem Grafikteil (mehr wollten und sollten sie auch nicht sein). Eine neue Richtung der Geldanlage, eine höchst riskante, aber dafür auch sehr spannende, ist seit dem 26. Januar 1990 auch für den Privatanleger gangbar: der Optionshandel. (Lesen Sie bitte hierzu auch die kurzen Ausführungen im Grundlagenartikel.)



Bild 1: Die Arbeitsmenüs für verschiedene Systemeinstellungen

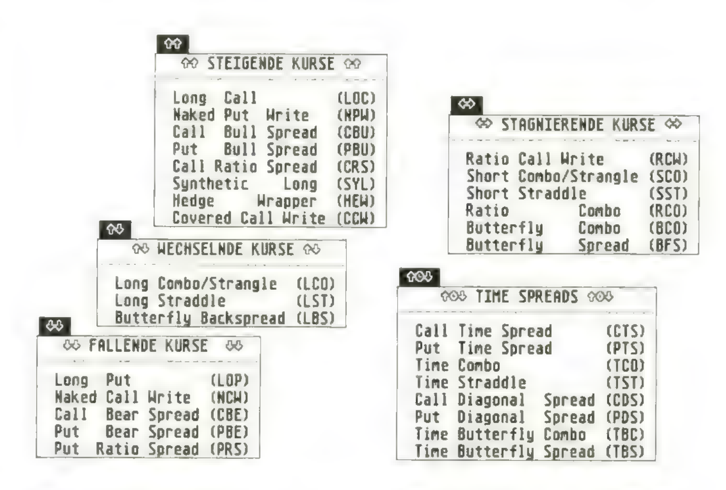


Bild 2: Die reichlichen Strategie-Menüs. Vielleicht etwas zu viel für die Desktop-Leiste?

Es hat nicht sehr lange gedauert, bis ein entsprechendes Programm (ausschließlich für Optionen) auf dem ATARI ST verfügbar war. Mit "CATO" soll ein 'Aufbruch in die neue Dimension der Kapitalanlage' (Original-Werbespruch) möglich sein. Um es aber gleich vorwegzunehmen: Der Autor dieser Zeilen hatte mangels Masse (damit ist das liebe Geld gemeint) leider nicht die Chance, den Anlageempfehlungen von CATO Folge zu leisten. Solch ein 'heißer Test' wäre wohl der beste Weg, die Leistungsfähigkeit von CATO abzuklopfen. So müssen wir uns hier auf Trockenübungen beschränken, die uns aber dennoch sehr anschaulich die Funktionsvielfalt des Programms aufzeigen.

Die Börse wartet

Ausgestattet mit einem ATARI ST-Computer (mit mindestens 1 MByte RAM, CATO benötigt davon mindestens 700 kB!) sowie einem Schwarzweißmonitor könnte es eigentlich losgehen. Obwohl CATO mit einem Diskettenlaufwerk auskommt, wäre eine Festplatte sehr zu empfehlen. Auch ein Drucker ist für die Ausgabe von Protokollen und Grafiken recht nützlich. Ein Laserdrucker ist leider noch nicht ansprechbar. Das Programm hat keinen Kopierschutz und läßt sich problemlos auf der Festplatte installieren.

Der erste Schritt nach dem Programmstart führt uns (etwas abseits liegend) in das Menü *Extras* zur Eingabe der sogen. Fixdaten. Dort wird z.B. festgelegt, mit welchem Anlagebetrag (Investment) man beabsichtigt einzusteigen, was für Gebühren in welcher Höhe anfallen usw. Diese Werte liegen dann allen späteren Auswertungen zugrunde. Bei jeglicher Änderung der Fixdaten läuft sofort eine Neuberechnung in den Strategiefenstern (betrachten wir uns gleich noch genauer) ab.

Als zweites müssen die 'technischen' Daten der Option im Menüpunkt *Optionen Eingabe* erfaßt sein. Hierfür stehen 16 Bildschirmseiten zur Verfügung die jeweils ein PUT und ein CALL mit allen wichtigen Einzelinformationen zu einer Aktie aufnehmen. Wahlweise kann auch die Übernahme aus einer schon bestehenden Liste (im RAM) oder von einer Datendiskette geschehen. Ein besonderes Bonbon ist das Anwählen einer sogenannten BTX-Datei, die ein spezielles Steuerprogramm (Software-Dekoder, separat zu erwerben) von Informationen aus dem Bildschirmtextsystem der Bundespost zusammengestellt hat. CATO empfiehlt zur automatischen Kurseinlesung den "BTX/Vtx-Manager" aus dem Hause Drews, Heidelberg. Anmerkung zu BTX: Es gibt derzeit keine preisgünstigere Art, tagesaktuelle Daten zu Optionen zu erhalten, als über Bildschirmtext.

Wenn diese Rahmendaten feststehen, kommt die eigentliche Arbeit von CATO: das Auswerten und Anzeigen verschiedener Strategien. Es stehen insgesamt 30 dieser 'Spekulationswege' offen. Natürlich sollte der Benutzer mit den grundlegenden Gesetzmäßigkeiten des Optionsmarktes vertraut sein. So gibt es auch nicht DIE optimale Strategie, sonst wären die restlichen 29 ja sinnlos. Allein der Anleger soll aus seiner Markteinschätzung und seiner Risikobereitschaft die optimale Strategie herausfinden. CATO kann ihm dabei nur Werkzeug sein - entscheiden muß der Anleger selbst.

Es würde an dieser Stelle zu weit führen, alle 30 Strategien vorzustellen und die Gegebenheiten für ihre Anwendung abzuprüfen. Wichtig ist uns nur: Wie werden die verschiedenen Strategien anschaulich gemacht?

Ein Bild sagt mehr

Wie schon an anderer Stelle festgestellt, kann selbst der erfahrungsreichste Börsenprofi mit einer Riesenliste von Zahlenkolonnen kaum etwas anfangen. Nichts ist übersichtlicher und aussagekräftiger als eine statistische Kurve. So kommen wir zu dem Haupttätigkeitsfeld von CATO: den Grafikfenstern.

Üblicherweise ist das Hauptarbeitsfenster in zwei Hälften geteilt. Dort können zwei unterschiedliche Strategien sogar zu zwei völlig verschiedenen Optionsscheinen nebeneinander verglichen werden. Dies macht durchaus Sinn! Oft ähneln sich Marktsituationen, so daß auch die Resultate der Strategien (für einen Titel) sehr ähnlich sein können. Andererseits ist es denkbar, daß sich ähnliche Titel (mit derselben Strategie) identisch entwickeln. Das Doppelfenster macht diese Vergleichsanalyse sehr gut sichtbar.

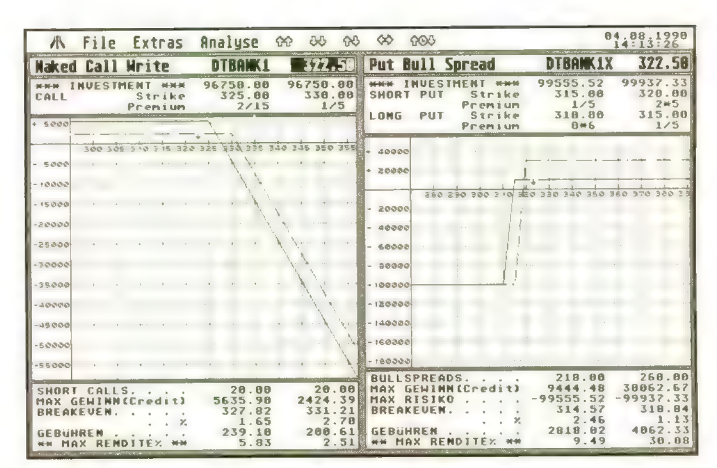


Bild 3: Typische Arbeitseinteilung mit zwei getrennten Strategiefenstern

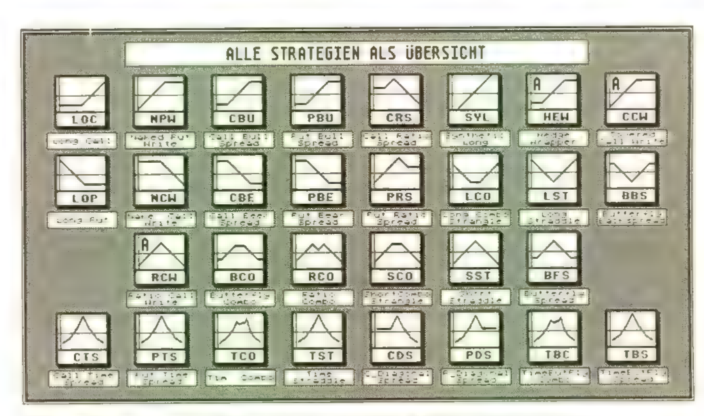


Bild 4: Die komplette Strategieübersicht mit Minidiagrammen

Bild 5: So sieht die Datenübernahme aus einer Tabelle aus.

File Extras Analyse — 04.08.1990 14:31:56

Titel: BMW_.1_ — OPTIONS-DATEN 04.06.1990 14:31 — Kurs: 322.50

Exp-Dat:17AUG (1.NORM, 1.TIME) — Exp-Dat:21SEP (2.NORM, 2.TIME) — Exp-Dat:19OKT (3.NORM, 3.TIME) — Exp-Dat:16NOV (4.NORM, 4.TIME)

Calls (Premium)				Strike	Puts (Premium)			
36/5	33/15	31*4	28*4	295.00	0*4	2*1	3/7	4*6
31/13	29*3	26*6	23/9	300.00	0*4	2*3	3/13	5/3
27/9	25.00	22*2	18*5	305.00	0/9	2*6	4*3	5/13
23*4	20*7	17/15	13/13	310.00	0*6	3/7	5*2	6/11
19/11	17/1	14/1	9*3	315.00	1/5	4*4	6*3	7*7
16*2	13*5	10*5	5/11	320.00	2*5	6/1	7*7	9/5
16NOV	19OKT	21SEP	17AUG	322.50	17AUG	21SEP	19OKT	16NOV
13/3	10*5	7*6	2/15	325.00	4*7	8*1	9*7	11/3
10*4	8/1	5*3	1/5	330.00	8*2	10*6	12*2	13/7
8/3	6/1	3/9	0/7	335.00	12*4	13/15	15*1	16/1
6*2	4/5	2/5	0*1	340.00	17*4	17*5	18*3	19/1
4/11	3/1	1*3	0/1	345.00	22*4	22*4	22*4	22*4
3/7	2/1	0/13	_____	350.00	27*4	27*4	27*4	27*4

Seite 1-16 [-] 1 [+] — STRIKE INTERVALL A 1 2 2.5 5 10 20 50 100 — FILE AUSWAHL (LADEN) DATENBLATT, SEITE.EIN, H & F.DOC — SPEICHERN ABBRUCH LADEN OK

Bild 6: Das Datenblatt für eine Aktie mit Kauf- und Verkaufsoption

Supergraph

Wen das stört, kein Problem: Die Funktion *Supergraph* läßt eine der Skizzen zur genaueren Betrachtung über den ganzen Bildschirm erscheinen. Sogar eine Überlagerung von maximal 4 Graphen ist in dieser Ganzseitendarstellung möglich. Die Funktion "Superposition" arbeitet fast so ähnlich. Bei manchen Strategien kann es sinnvoll sein, die Varianten in einem Bild zusammenzufassen. Bei *Superposition* werden zwei Strategievarianten zu einer Optionsposition zusammengerechnet. Hier führt CATO also eine Addition bzw. Subtraktion der Graphen durch und zeichnet das Ergebnis als dickere Linie ein.

Automatik starten

Es passiert wahrscheinlich nicht gerade selten, daß man bei der Fülle der Strategien 'vor lauter Bäumen den Wald nicht sieht'. Eine Strategieautomatik kann nun auf einen Optionstitel angewandt werden, so daß CATO alle 30 Variationen hintereinander durchrechnet. Da CATO diese dabei auch gleich mitzeichnet, sieht man auf dem Bildschirm in Windeseile die Diagramme vorbeiblitzen. Das bringt dem Zuschauer leider nichts. Weit interessanter ist das Resultat: Es wird eine Rangliste ausgegeben, die entsprechend der statistischen Durchschnittsrendite sortiert ist (Anmerkung für Kenner: "Kainer-Hofbeck-Modell"). Die derzeit effektivste Strategie steht demnach an erster Stelle. Auch der zu erwartende Gewinn/Verlust ist für jede Strategie abzulesen. (Wertung: sehr nützlich)

Die Strategien

Man kann die Taktik der Vorgehensweise in vier Grundtypen aufteilen, je nachdem, welche Markttendenz für die Aktien erwartet wird:

- steigende Kurse
- fallende Kurse
- wechselnde Kurse (also starke Schwankungen in beide Richtungen) und
- stagnierende Kurse (kaum Bewegung).

Für die hohe Schule der Optionen ist eine fünfte Strategiefamilie eingebaut: die "Kalenderblätter" (engl. time spreads). Diese Berechnungsart unterscheidet sich sehr stark von den genannten vier 'einfacheren' Typen, und es sollten sich wirklich nur absolute Kenner auf diesen schwierigen Pfaden tummeln. Die Berechnung dieser Strategien setzt voraus, daß nach Laufzeitende von kürzer laufenden Optionen die länger laufenden verkauft werden. (Anmerkung für Kenner: Die Restwertbestimmung zum Verfalltag und damit die Berechnung bzw. zeichnerische Darstellung von Gewinn oder Verlust erfolgt auf dem "Black-Scholes-Modell"). Gewinn oder Verlust können sich bei ungenauer Wahl der Eingabeparameter extrem verschieben. Deswegen sollten alle technischen Daten der Option sorgfältig nachgeprüft werden - alles in allem eine sehr heikle Angelegenheit. Für den Laien erscheinen die Namen der Strategien als sehr belustigend: "Butterfly Backspread", "Short Combo Strangle", "Synthetic Long", um nur einige zu nennen.

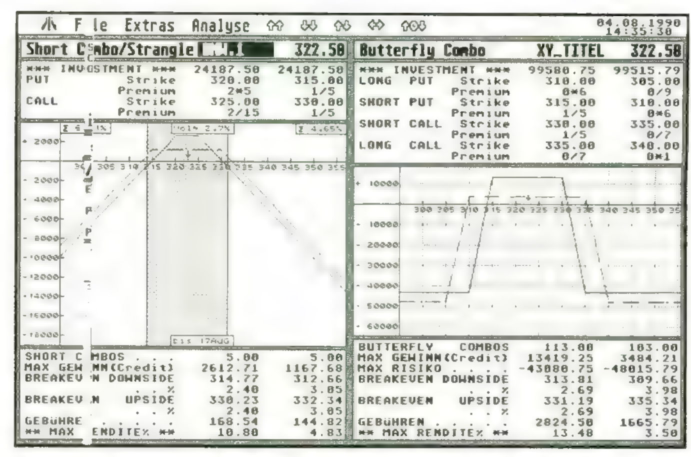


Bild 7: In das linke Diagramm ist die Schwankungsbreite der Aktie eingezeichnet (grauer Balken).

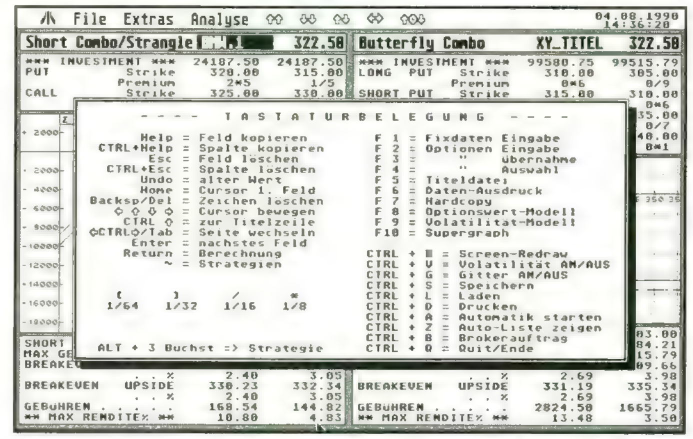


Bild 8: Auch per Tastenkombinationen läßt sich CATO bedienen.

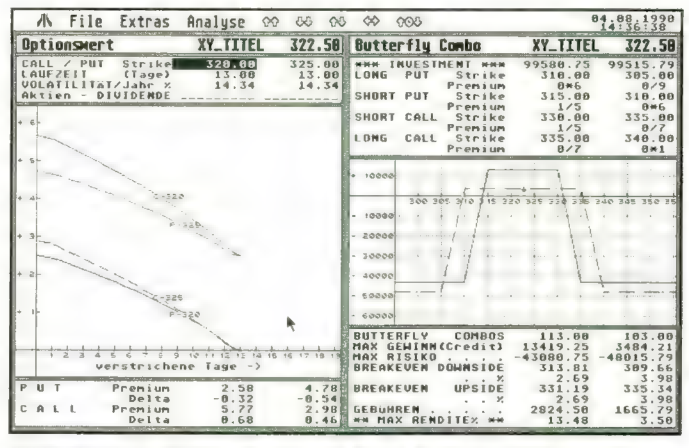


Bild 9: Die linke Grafik zeigt sehr schön den Zeitwertverfall einer Option.

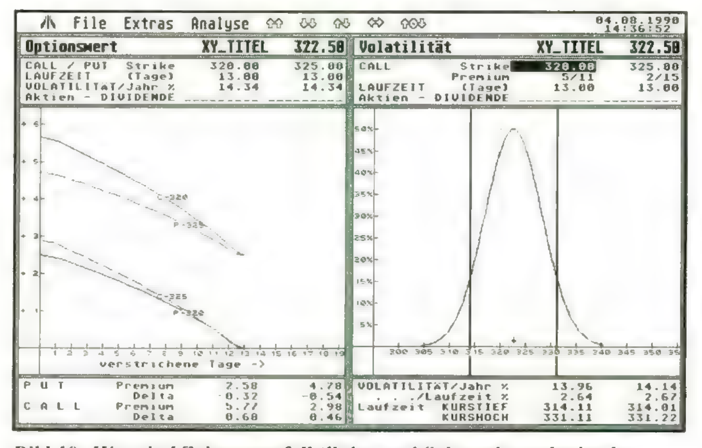


Bild 10: Hier sind Zeitwertverfall (links) und Schwankungsbreite der Aktie (rechts) gegenübergestellt.

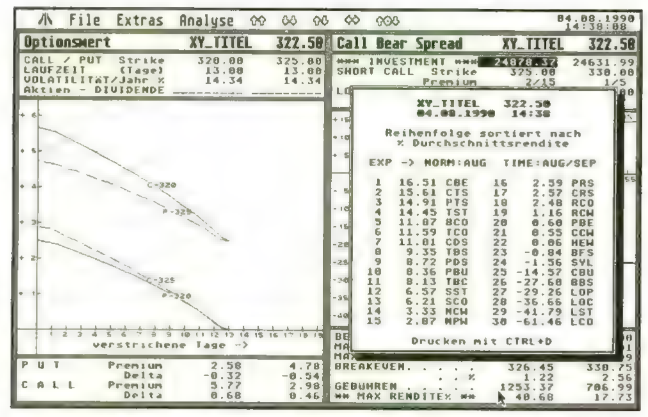


Bild 11: Ein Superbonbon von CATO: Rechts im Bild ist eine Vorschlagsliste aller Strategien zu sehen, nach Erfolgsaussicht sortiert.

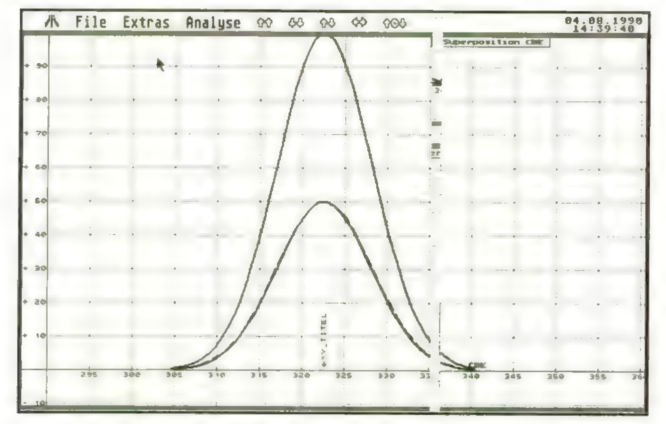


Bild 12: Zwei Schwankungsbreiten in der Supergraph-Darstellung.

Besonders die Datenübernahme aus dem günstigen BTX-Netz der Bundespost hat mir sehr gut gefallen. Zu beachten ist aber, daß die meisten Anbieter im Bildschirmtext für das Bereitstellen von aktuellen Kursdaten ihrerseits Gebühren verlangen. Hier sollte man unbedingt die Kosten vorher vergleichen. Schauen Sie mal bei folgenden Anbietern nach:

- Hornblower & Fischer *55553#
- Harald Beer *37717#
- Portfolio Concept *26161#
- Hartmut Schmidt *47110#

Ebenfalls sinnvoll ist das Auslösen aller Programmfunktionen per Tastenkombination (Maus in die Falle), denn die Menüs sind mir ein klein wenig zu überladen. Der Grafikaufbau geschieht sehr schnell, skaliert sich automatisch und ist, trotz kleiner Beschriftung, sehr gut zu interpretieren.

Zusammenfassend

CATO ist ein umfangreiches Programmpaket zur Optionsanalyse. Zeitungen wie “Wirtschaftswoche” und “Handelsblatt” sind voll des Lobes. Mit großem Bedauern muß der Autor dieser Zeilen (zu seiner eigenen Schande) bekennen, daß er nicht die Möglichkeit hatte, CATO “live” zu testen. Wer hat schon mal 20.000 DM (Mindesteinsatz) übrig, um zu beweisen, daß ein Programm richtig rechnet? Selbst o.g. Zeitungen haben das nicht getan. So muß ein ‘Trockentester’ von dem ausgehen, was er an ähnlichen Programmen kennt, und im ‘freien Spiel der grauen Zellen’ mutmaßen, wozu ein solches Programm fähig sein könnte.

Wenn man das Handbuch als Qualitätsmaßstab herannimmt, wird sich CATO im Alltagseinsatz sicher bewähren. Die annähernd 250 Seiten in einem handlichen DIN A5-Ordner erläutern sehr ausführlich den Optionshandel im allgemeinen und die einzelnen Menüpunkte und Funktionen von CATO im speziellen. Sehr großen Raum nimmt mit ca. 100 Seiten die Erklärung aller 30 Strategien ein. Eine klare Gliederung, reichlich Bilder und eine verständliche Sprache tun ihr übriges. Nur stellenweise sind mir die Erklärungen doch etwas zu langatmig.

CATO kostet 1298,- DM. Da habe ich einen bekannten Bankangestellten gefragt und er antwortete mir: “Wer sich als ernsthafter Anleger mit Optionen befaßt, bewegt sich in solch hohen Dimensionen, daß dieser Preis sicher nicht zu hoch ist.” Ob sich seine Bank auch einmal für CATO entscheiden würde, wollte er nicht mit Bestimmtheit sagen.

DK

Bezugsquelle für Händler:

H. Richter Distributor
Hagener Straße 65
5820 Gevelsberg

Informationen zum Programm:

knowledge EDV-Anwendungen
Beskidenring 11
8858 Neuburg-Donau



"Beast 2" wird ein intensiveres Spielgefühl bieten als "Beast 1". Das Spiel bietet eine vollkommen neue Scrolltechnik, die es erlaubt, in fast unmöglich vielen Richtungen von Screen zu Screen zu scrollen - in jeder Geschwindigkeit und bei 50 Bildern pro Sekunde! Außderdem ist jedes Monster individuell und besitzt "Intelligenz". Anders als in "Beast 1" muß man hier nicht alles und jedes abstechen, was einem über den Weg läuft, sondern man muß mit den Charakteren sprechen, und in einigen Fällen helfen sie dem Spieler auch, das Spiel zu beenden.



Beast 2

"Sunny Shine on the Sunny Side of Life" heißt das Debutspiel des Zigarettenherstellers Philip Morris. In diesem Adventure bestimmt der Spieler die Geschicke von Sunny Shine, einem talentierten Werbegrafiker, der verzweifelt Geld auftreiben muß, um ein zerstörtes Luxusauto zu ersetzen. Das Abenteuer führt ihn in eine Fernsehshow, an den Strand und nach Los Angeles. Andauernd tauchen Werbelogos für die Sargnagelmarke "L&M" auf. Etwas zu aufdringlich, wie ich meine.

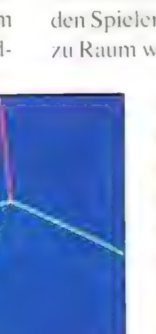
Gemetzel zu Pferde, blutige Schlachten mit Bogenschützen und gepanzerten Reitern: In "Dynasty Wars" herrscht Blutfehde zwischen zwei chinesischen Geschlechtern, die sich bis aufs Messer bekämpfen. Auf einem stattlichen Gaul galoppieren Sie durch chinesische Landschaften und prügeln mit Lanze oder Schwert auf alle Angreifer, die sich in den Weg stellen. Sprites und Hintergrundgrafiken sind hervorragend gezeichnet und absolut farbenprächtig. Leider scrollt der Bildschirm viel zu langsam und die Kollisionsabfrage ist ein wenig zu ungenau.

"Murders in Space", ein neues Detektivspiel von Infogrames, soll im September erscheinen. Wir schreiben das Jahr 2005 und Sie befinden sich auf einer Orbitalstation. Alles wurde wunderbar programmiert, so daß die Operation ein Erfolg ist... Allerdings wird es merkwürdig, als ein Mordversuch stattfindet, der die Crew der Station in Schrecken versetzt und die Verantwortlichen alarmiert - "Pegasus Space Consortium". Nun liegt es an Ihnen, einzuspringen und das Leben von 8 Mitgliedern der Crew zu retten. Sie müssen schnell handeln, der Countdown läuft. "Murders in Space" baut auf animierten Dialogen auf und beinhaltet exzellente Simulationen: manuelle Steuerung der Station und vieles mehr. Details werden herrlich genau und perfekt realistisch dargestellt.



Murders in Space

"Alpha Waves - The Dream Generator" ist nicht nur ein Spiel, sondern auch ein neues Konzept, sozusagen "emotionelle Software", eine Mischung aus Ideengeist und Technologie. Sinn des Spiels ist es, ein Teil in einem Würfel in drei Dimensionen handzuhaben, um damit eine magische Tür zu erreichen. Natürlich existieren in dieser dreidimensionalen Welt auch Wesen mit Intelligenz und einer fremden Sprache. Harmonischer Sound begleitet den Spieler, während er von Raum zu Raum wandert. Als erstes Spiel ermöglicht "Alpha Waves" die Kommunikation zwischen dem ST und dem Menschen (erscheint ungefähr Oktober von Infogrames).



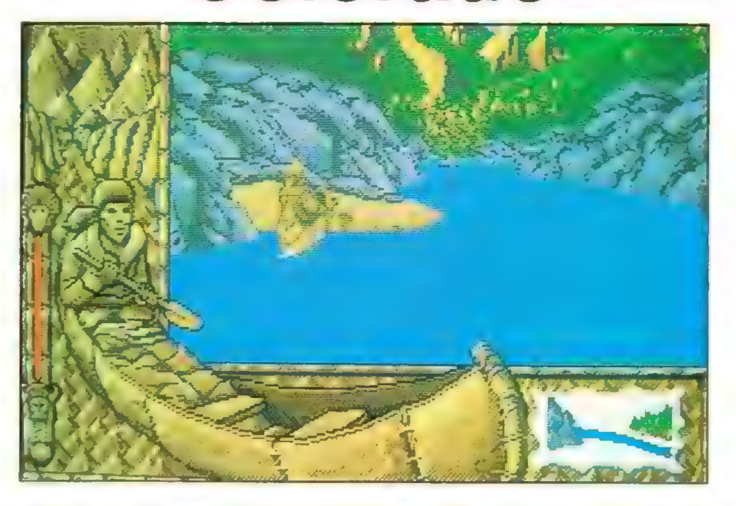


Alpha Waves

Seit neuestem hat Rainbow Arts eine eigene Hotline. Wer 0211/596761 wählt, kann ein Tonband abhören, das über Erscheinungsdatum, Preis und Besonderheiten neuer Rainbow Arts Spiele Auskunft gibt. Mit dabei sind auch Informationen aller an Rainbow Arts angeschlossenen Software-Häuser. Und das sind momentan: Lucasfilm Games, Broderbund, Innerprise, Reline und SSI.

ddf / MP / CBO

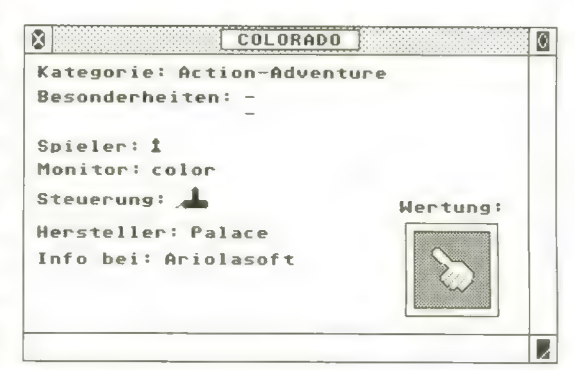
Colorado



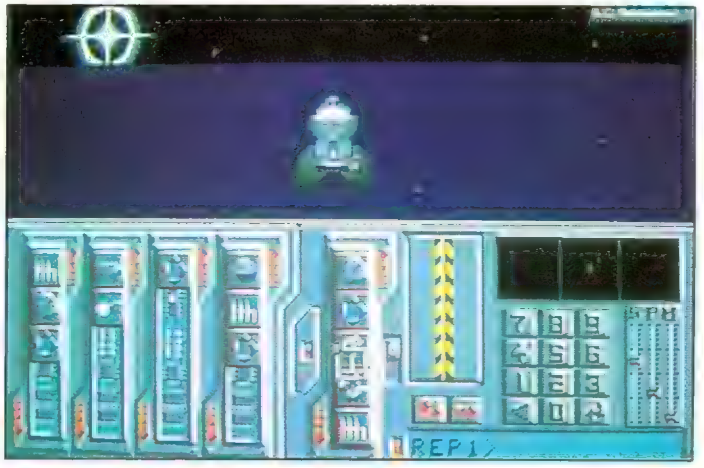
Wir schreiben das Jahr 1801, irgendwo in Colorado. Trapper David O'Brian kommt einem alten Cheyenne-Indianer zu Hilfe, den gerade die wilden Pawnies überfallen. Mutig vertreibt er die Peiniger, kann den Tod des alten Häuptlings aber nicht mehr verhindern. Bevor der Cheyenne in die ewigen Jagdgründe entschwebt, drückt er dem Abenteurer eine Karte in die Hand. Sie zeigt den Weg zu einer geheimnisvollen Goldmine, wo unermeßliche Reichtümer ruhen. Vom Schatzfieber gepackt, macht sich O'Brian auf die Suche. In Silmarils' neuem Action-Adventure übernimmt der Spieler erwartungsgemäß die Rolle des Trappers und versucht, sich den Gesetzen des Wilden Westens anzupassen. Die Regel lautet: töten, um nicht selbst getötet zu werden. So besteht ein Großteil des Spiels darin, die Gegner (Indianer, Banditen, wilde Tiere...) durch gezielte Schüsse oder Axthiebe auszuschalten. Mit etwas Glück lassen die Feinde nach ihrem Ableben einen Gegenstand zurück, der O'Brian im weiteren Verlauf des Geschehens nützlich ist.

Das Kanu außer acht lassend, macht er sich zu Fuß auf den Weg. Sobald er den Bildschirmrand erreicht, wechselt die Hintergrundgrafik, was einige Zeit dauert und den Spielfluß unterbricht. Ein ordentliches Scrolling wäre hier angebracht. Vorsicht in Ufernähe oder an steilen Berghängen! Dank der ungenauen Steuerung stürzt man leicht in die Tiefe. Der geringste Fehltritt beendet das Spiel. Schon zu Beginn stößt O'Brian auf die ersten Gegner, die noch kein großes Hindernis darstellen. Der Schwierigkeitsgrad steigt jedoch schnell an, da immer mehr Treffer nötig sind, um einen Feind außer Gefecht zu setzen. Mit jedem Treffer fällt die Energieanzeige des Trappers, bei Null ist das Spiel logischerweise beendet. Wer sich mit der nervigen Steuerung vertraut gemacht hat und über die ersten Bilder hinauskommt, wird mit prächtigen Grafiken en masse belohnt. Einfach wunderschön! Das kann ich von den Animationen allerdings nicht behaupten. Schade! "Colorado" besitzt viele kleine Schwächen: ungenaue Steuerung und schwache Animationen. Nur wer darüber hinwegsehen kann, erlebt ein unterhaltsames Wild-West-Abenteuer.

CBO



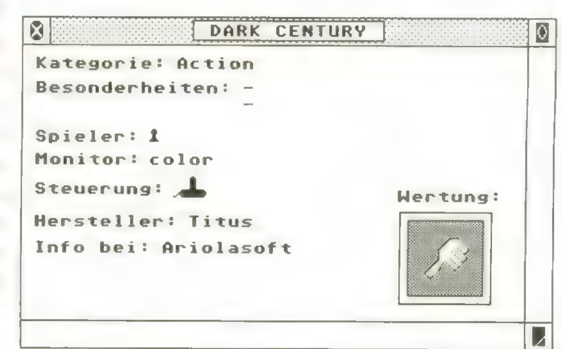
Dark Century



Düstere Zukunftsaussichten: Schenkt man den Story-Schreibern von "Titus" Glauben, so nimmt die Kriminalität im Jahre 2290 erschreckende Ausmaße an. Alle Gefängnisse sind überfüllt, deshalb hält die Justiz viele Verurteilte in riesigen Satelliten gefangen, die irgendwo im Weltraum treiben. Dort verrichten die Knasties Zwangsarbeit unter allerschwersten Bedingungen. Zur Bewachung der Schwerverbrecher patrouillieren Aufseher mit riesigen Panzern. Falls ein Gefangener versucht, zu fliehen, wird damit die Verfolgung aufgenommen. Dummerweise sind ein paar Gauner in den Besitz einiger Kampfpanzer gelangt. Mit aller Entschlossenheit versuchen sie zu entkommen, und die korrupte Regierung zu stürzen. Ein Kampf auf Leben und Tod beginnt!
Eine wirklich spannende Hintergrundgeschichte -wenn doch bloß das Spiel nicht so primitiv und langweilig wäre. Die Aufgabe des Spielers besteht darin, die feindlichen Panzer aufzuspüren, um sie mit gezielten Feuerstößen zu vernichten. Zwischendurch schaut man auf den Scanner, der sämtliche Objekte anzeigt. Da gibt es verbündete Panzer, Gegner und Energieblasen. Letztere bieten die Möglichkeit, Energie aufzufrischen. Deshalb sollte man im Eifer des Gefechts unbedingt darauf achten, daß man die Blasen nicht versehentlich abschießt. Wer sich nicht im Automatik-Modus befindet, sondern die Steuerung seines Panzers selbst übernimmt, wird auf den Scanner nicht verzichten können. Damit spürt der Panzerpilot seine Gegner auf und feuert, bis sie in alle Einzelteile zerberbsten. Daraufhin sucht man ein neues Opfer, das auf die gleiche Weise ins Jenseits befördert wird.

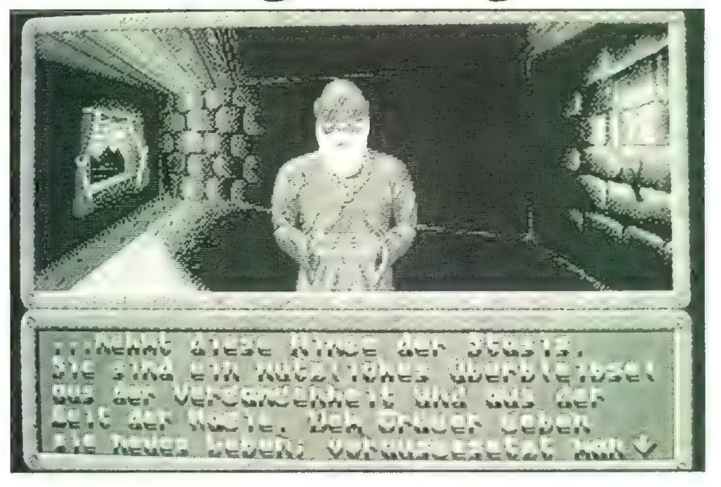
Wer sich allerdings lieber gemütlich im Sessel zurücklehnen und die Kämpfe in aller Ruhe beobachten möchte, programmiert die Bordcomputer. Es ist möglich, jedem Panzer mit Hilfe von Icon-Befehlen ein eigenes Programm zu verpassen. Damit suchen die Kampfmaschinen ihre Gegner selbständig und versuchen, diese durch geschickte Manöver auszuschalten. Die Darstellungen von Panzer und Planetenoberfläche sind mehr als kläglich. Wer auf der Suche nach einer gelungenen Panzersimulation ist, sollte um "Dark Century" einen großen Bogen machen.

CBO



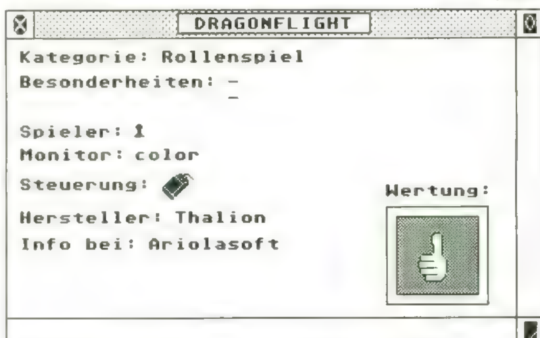
Dragonflight

Vor vielen, vielen Jahren bevölkerten Drachen die Welt. Diese riesigen Kreaturen waren friedfertig und äußerst intelligent. Sie beherrschten die hohe Zauberkunst. Doch trotz dieser magischen Fähigkeiten hatten die Feuerspeier ein Problem: sie waren unfruchtbar. So ist die Ausgangssituation des neuen Rollenspiels "Dragonflight". Die Aufgabe des Spielers besteht nun darin, Zaubersprüche zu finden, die Magie zu erlernen und das Geheimnis der ausgestorbenen Drachen aufzuklären. Wie bei Rollenspielen üblich, bestimmt der Spieler nicht die Geschicke eines einzelnen Helden, sondern die einer ganzen Gruppe. In "Dragonflight" besteht sie aus zwei Menschen, einem Zwerg und einer Elfe. Mit dieser Truppe muß der Spieler alte Dungeons erforschen, Tempel und Städte besuchen sowie Seefahrten zu anderen Inseln unternehmen. Während der Reise begegnet man den unterschiedlichsten Figuren. Menschen, die wertvolle Informationen preisgeben, oder Monster, die gleich angreifen. Vor der Reise in die Wildnis sollte man in den Städten Ausrüstung, Waffen und Proviant organisieren. Bezahlt



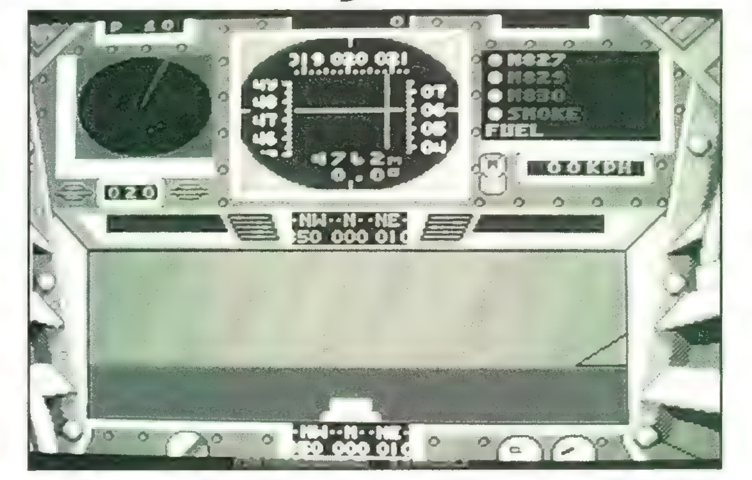
wird mit purem Gold. Leider sind die vier Abenteurer damit nicht besonders gesegnet. Deshalb müssen sie alte Schatzkisten auftreiben, die häufig bewacht werden, auf den Wegen dorthin lauern wilde Tiere. Wer eine Kiste findet, darf sich über Gold, Zaubersprüche, magische Tränke und Edelsteine freuen.
Den Schauplatz eines Fights zeigt der Computer stets in einer Seitenansicht. Dazu wird ein kleiner taktischer Bildschirm eingeblendet, der die Position aller Beteiligten aus der Vogelperspektive zeigt.
Durch die so gewonnenen zwei Perspektiven ist es leicht möglich, einen Gegner zu umzingeln und von zwei Seiten niederzuknüppeln. So schön ist der Kampf bisher bei keinem anderen Rollenspiel realisiert worden. Jede Phase ist voll animiert. Nicht nur bei der grafischen Darstellung, sondern auch an der Steuerung haben die Programmierer kräftig gefeilt. Fast alle Aktionen werden über Icons gesteuert. Durch den klaren Aufbau ist die Einarbeitungszeit erfreulich kurz. Was die Spielbarkeit angeht, ist "Dragonflight" geradezu vorbildlich und in Sachen Grafik und Sound meisterhaft .Wer Rollenspiele mag und schon immer eins gesucht hat, das neben Komplexität und Spielbarkeit auch noch durch ausgereifte Programmiertechnik begeistert, sollte sich Thalions Fantasy-Erlebnis unbedingt zulegen.

CBO



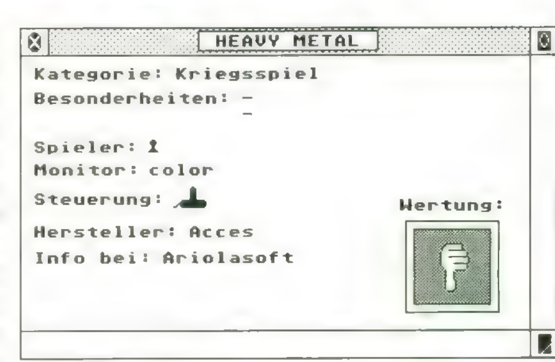
Heavy Metal

Allen Abrüstungs- und Friedensverhandlungen zum Trotz bietet dieses Spiel die Möglichkeit, sich mit dem Feind in drei der stärksten Kampffahrzeugen der Welt zu messen.
Mit dem Rang eines Kadetten beginnend, versucht der Spieler, sich an der "Officers Trainings School " zu qualifizieren.
Als Führer des ADAT besteht seine Hauptaufgabe in der Überwachung und Säuberung des Luftraums. Nicht weniger schwer ist die Beherrschung des MBT-Kampfpanzers.Der untrügliche Beweis eines verlorenen Panzers (gesprungene Sichtscheibe) läßt nicht lange auf sich warten. Am lebendigsten wirkt die Sequenz des FAV-Einsatzes. Mit einem Dünenbuggy rast man durch die feindlichen Gefilde und wird dabei mit den unterschiedlichsten Hindernissen konfrontiert. Dauerfeuer einschalten, und ab geht die Post! Hat es der aufstrebende Soldat zum Leutnant gebracht, kann er als vollwertiges Stabsmitglied im zweiten Teil des Spiels, der taktischen Befehlszentrale, auf einer Lagekarte einsetzen.

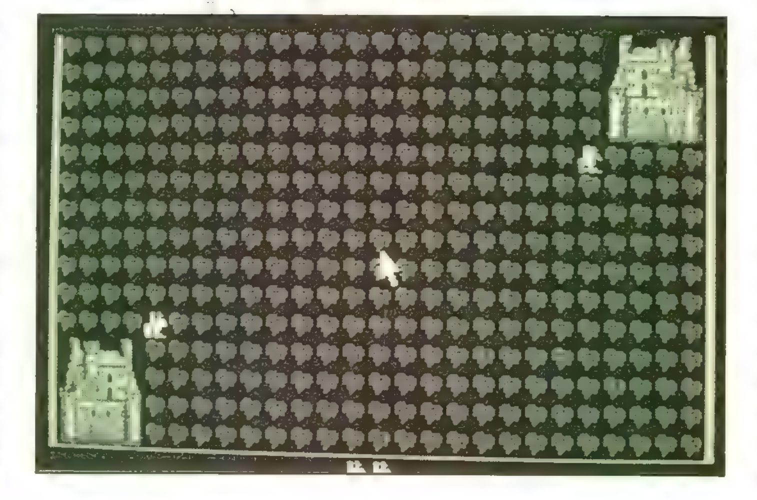


Hier werden alle Optionen des Spiels vereint, auch die anfangs zu absolvierenden Action-Sequenzen. Der Schwerpunkt liegt allerdings in den taktischen Bereichen Truppenaufstellung, Kampfhandlungen und Nachschub. Der Spieler muß versuchen, durch geschickten Einsatz der Einheiten ein Vordringen des Gegners zu verhindern. Mit Hilfe einer übersichtlichen Karte trifft er zunächst die erforderlichen taktischen Entscheidungen. Sind Marschrichtungen und Befehle (Angriff, Rückzug etc.) ausgewählt, kann der K(r)ampf beginnen. Wer direkt ins Kriegsgeschehen eingreifen will, wählt die gewünschte Option aus und befindet sich sofort im Bombenhagel an vorderster Front.
Neben der unpopulären Spielidee hat HEAVY METAL auch sonst nicht viel zu bieten. Die Grafiken und Sounds sind zwar teilweise recht gut gelungen, jedoch fehlt es dem Game an Atmosphäre und Abwechslung. Wer glaubt, seinen Spielspaß eventuell im taktischen Teil des Programms zu finden, wird enttäuscht sein.

CBO



Ritter



Wir befinden uns im Jahre 1472. Schauplatz des Geschehens ist ein kleines, unerschlossenes Tal in Britannien. Lediglich zwei angrenzende Burgen sind die letzten Zeugen vergangener Zivilisationen. Um die Gegend ein wenig zu beleben, hat der Herrscher des Landes einen kleinen Wettkampf zwischen seinen beiden Günstlingen ausgeschrieben. Der Sieger bekommt das Tal als Lehen auf Lebzeiten überschrieben.

Ziel des Wettstreites ist es, das in Parzellen (Felder) aufgeteilte Tal durch strategisch geschickte Bewirtschaftung teilweise oder ganz für sich zu gewinnen und die Burg des Widersachers zu zerstören.

Der Spielablauf: Die Spieler roden, zivilisieren und setzen alles daran, den Wert der Parzellen zu erhöhen. Aus Rodung wird Hütte, aus Hütte Haus usw. Wenn ein Gebiet von 3x3 Feldern kolonisiert wurde, kann ein Kastell errichtet werden.

Nun ist es möglich, die heißersehnten Steuergelder einzutreiben und neue Ritter zur Verstärkung anzuwerben. Je mehr sich auf den Handel einlassen, desto größer ist die Chance, die gegnerischen Felder und letztendlich auch die Burg zu erobern.

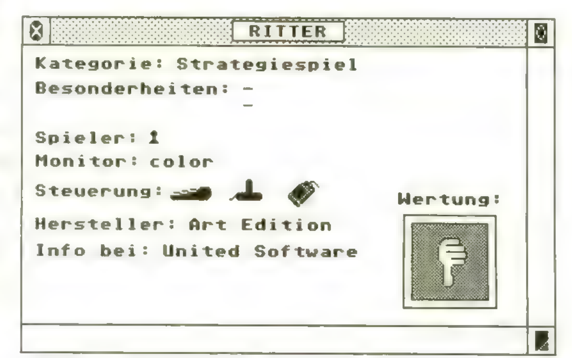
Diese Kastelle sind zwar schwer einzunehmen, doch wenn durch entsprechende Belagerung die Mauern erst einmal beschädigt sind, dauert es nicht lange, bis die dort heimischen Bewohner davonziehen. Nun muß das brachliegende Land neu bearbeitet werden, um für eine erneute Zivilisation vorbereitet zu sein.

Wurde die feindliche Burg erobert, bekommt man einen Platz in der Ahnengalerie und in der Highscore-Liste, die auf Diskette abgespeichert wird.

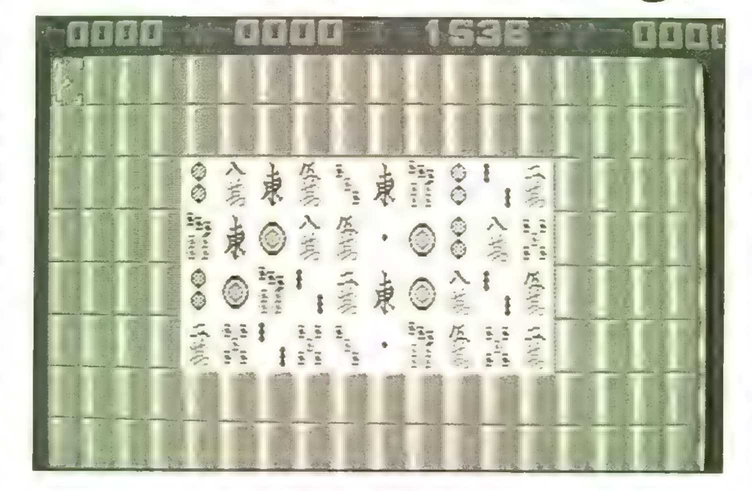
Die Steuerung des Spiels erfolgt wahlweise über Maus, Joystick oder Tastatur. Man spielt entweder zu zweit oder allein gegen den Computer, der sich seltsamerweise immer die rechte Burg aussucht. Leider hält das Spiel nicht das, was die Verpackung verspricht. Angepriesen als rasantes Strategiespiel, das viel Vergnügen und Kurzweil garantieren soll, kann RITTER weder durch Grafik oder Sound (sofern irgendwo im Speicher welcher existiert) noch durch gelungene Animation überzeugen.

Ein dermaßen schwaches Programm als Vollpreisspiel anzubieten, ist schon eine Unverschämtheit.

CBO


RITTER
Kategorie: Strategiespiel
Besonderheiten: –
Spieler: 1
Monitor: color
Steuerung:
Hersteller: Art Edition
Info bei: United Software
Wertung:

Lin Wu's Challenge



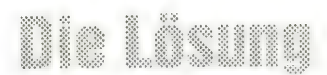
Als Pate für "Lasersofts" Tüftel-Game standen mehrere Gesellschaftsspiele wie "Mah Jongg", "Memory" und auch der Spielautomat "Match it". Es geht darum, auf einem bambusfarbenen Brett angeordnete Spielsteine paarweise abzuräumen.Ziel des Spiels ist es, die insgesamt sechzig Bretter innerhalb einer vorgegebenen Zeit komplett zu leeren. Hat sich der geschickte Stratege bis dorthin vorgearbeitet, trifft er im Endspiel auf den großen Meister Lin Wu höchstpersönlich. Ist auch diese Hürde genommen, kann das Spiel fast unendlich lange fortgesetzt werden, denn jetzt übernimmt der Computer das Zusammensetzen neuer Bretter. Wer glaubt, das Spiel sei deshalb leichter zu meistern, täuscht sich gewaltig. Die Regeln sind zwar schnell erklärt, die Unterschiede zwischen Theorie und Praxis aber nötig: Ein Steinpaar läßt sich nur entfernen, wenn beide Teile mit einer Linie verbunden werden können, die maximal zwei Ecken umfaßt. Zusätzliche Hilfsfunktionen erhöhen die Spielbarkeit und vor allem die Motivation. Allerdings stehen sie nur zur Verfügung, wenn genügend "Bars" gewonnen werden. Die "Show" Funktion bewirkt, daß durch Drükken der Help-Taste ein möglicher Zug angezeigt wird. Mit der M-Taste ("Move") besteht die Möglichkeit, einen Stein zu verschieben, um sich auf diese Weise in eine bessere Ausgangslage zu versetzen. Die Steuerung des Cursors erfolgt wahlweise per Joystick oder über Tastatur. Lobenswerterweise verfügt "Lin Wu's Challenge" auch über einen Two-Player-Mode, der beide Ports berücksichtigt. Grafisch ist das Game ganz ansprechend gemacht. Alle Muster auf den Steinen sind gut zu erkennen. Ermüdungserscheinungen wie bei "Turn it" sind damit ausgeschlossen. Einziges Manko: Man ist gezwungen, immer mit der Originaldiskette zu spielen, ein lauffähiges Backup gibt es für 10 Mark beim Hersteller. Davon einmal abgesehen ist Lin Wu's Challenge ein prima Denkspiel, das in puncto Präsentation seinesgleichen sucht. Die Verpackung besteht nicht etwa aus ödem Plastik, sondern aus stilgerechtem Chinaholz. Auch das vergilbte Anleitungspapier paßt hervorragend zum Ambiente. Wer gerne spielt und dabei denkt, sollte sich Lin Wu's Challenge unbedingt einmal zu Gemüte führen. Vorsicht Suchtgefahr!

CBO

LIN WU'S CHALLENGE
Kategorie: Denkspiel
Besonderheiten: –
Spieler: 1...11
Monitor: color
Steuerung:
Hersteller: Lasersoft
Info bei: Ariolasoft
Wertung:

Larry III

denn aller guten Dinge sind drei



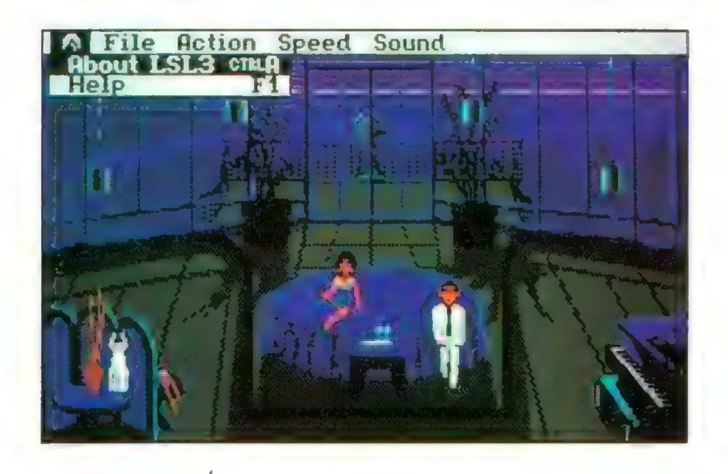


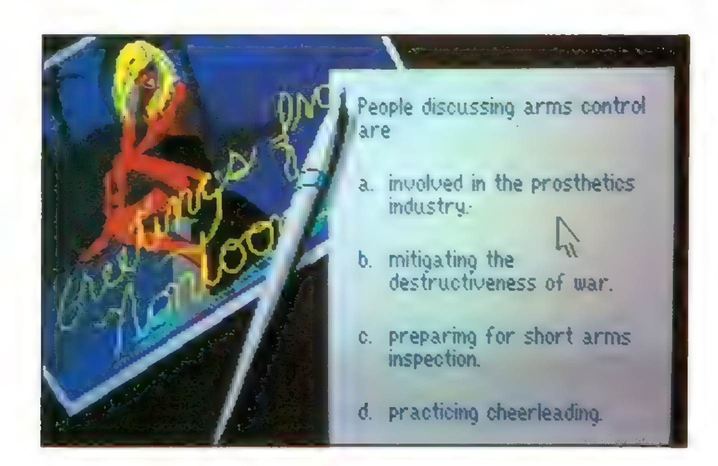


SIERRA On-Line sei Dank: Unser Junggeselle Larry Leisure befindet sich noch immer auf der Suche nach der Frau für's Leben und schenkt uns erneut stundenlangen Spielspaß. Das Adventure, dem es auch diesmal nicht an tollen Spielideen mangelt, wird durch ansprechende Grafiken und durchgehende musikalische Untermalung zur Konkurrenz für so manchen Kino- oder Fernsehfilm. Die SIERRA-Programmierer legten diesmal den Schwerpunkt eindeutig auf die Hintergrundmusik, die unter Verwendung des ROLAND MT-32 Sound-Moduls oder des neueren CM-32L-Moduls schnell zum Hauptgrund des Spielens werden kann.

Geendet hatte bei Larry II alles in schönster Harmonie: Larry durfte als Retter von Nontoonyt-Island die wunderschöne Tochter des Inselhäuptlings namens Kalalau ehelichen. Beide lebten glücklich und zufrieden in einem wunderschönen Haus auf einer wunderschönen Insel - vielleicht zu zufrieden? Denn mit zunehmendem Bauchumfang von Larry fühlte sich Kalalau immer mehr zu einer ehemals kanibalischen, Harley-fahrenden, Einarmige-Banditen reparierenden Frau hingezogen, was mit einem Rausschmiß Larry's aus dem Hafen der Ehe führte. Danach war ihm sein Job bei Kalalau's Vater natürlich auch nicht mehr sicher - aber was sind Sicherheiten gegen das wunderbare Gefühl wiedergewonnener Freiheit. Wieder ganz der Alte nimmt Larry am Strand gleich Kontakt auf zu einer wunderschönen, langbeinigen Blondine, mit der er dank der in seiner Postbox gefundenen Kreditkarte schnell zur - nicht gerade jugendfreien - Sache kommt. Eine Krabbe bereitet dem Vergnügen ein jähes Ende - zurück bleibt neben Schmerzen lediglich ein Messer, mit dem Larry, nachdem er es an der Casino-Treppe geschärft hat, aus einem gefundenen Granadilla-Holzstück ein ansehnliches Island-Souvenir schnitzt. In den hoteleigenen Cabanas tauscht Larry seinen berüchtigten weißen Polyester-Leisure-Suit gegen einen selbstgewebten Grasrock aus und bewegt sich erneut gen Strand, diesmal als Souvenir-Verkäufer mit mehr Erfolg. Ein erneuter Kleiderwechsel und das am Strand zurückgelassene Handtuch von Tawni bringen die ersehnte frauenantörnende Hautfarbe. In die Casino-Show kommt Larry durch eine kleine Bestechung und durch Vorzeigen seiner Karte in Form der persönlichen "Nontoonyt-Tonite"-Ausgabe. Nach der Show erfährt er bei einem Gespräch mit Cherri, dem Star des Abends, daß er sie mit seinem Stück Land glücklich machen könnte. Bei seinen Anwälten regelt er alles und bekommt dann letztendlich - nach kurzem Verlassen des Büros - vom sehr beschäftigten Roger die gewünschte Bescheinigung. Von Cherri bekommt er dafür eine heiße Belohnung, die mit einem Tanz auf der Bühne und vielen Dollarscheinen endet. Larry rennt mit dem Geld sofort zu seiner Anwältin, wo ihn aufgrund seines weiblichen Outfits neben der langersehnten Scheidungsvereinbarung inklusive Fat-City-Mitgliedskarte wieder ein heißes Abenteuer erwartet. Beim Umdrehen der Karte entdeckt er neben der Schranknummer einen Hinweis auf dessen Kombination (jeweils die Seitenzahl der Anzeigen im "Nontoonyt-Tonite") und beschließt, seinem Körper einen etwas jugendlicheren Touch zu geben. Dank diverser Body-Building-Übungen und anschließender Dusche mit ausschweifender

Einseifung (gottseidank hatte er die Seife von der Cabana mitgenommen!) gelingt ihm dies überaus erfolgreich, wie ein Besuch bei Bambi im Aerobic Studio zeigt (help Bambi with video). Zur Entspannung hilft danach nur noch ein Ab- unzensiert seinen Sinn des Lebens. Allerdings immer mehr auf Kosten der berühmt-berüchtigten Larry-Dialoge, die zeitgemäßen optischen Anzüglichkeiten weichen mußten. Aber dann kommt Patti, und mit ihr wieder die

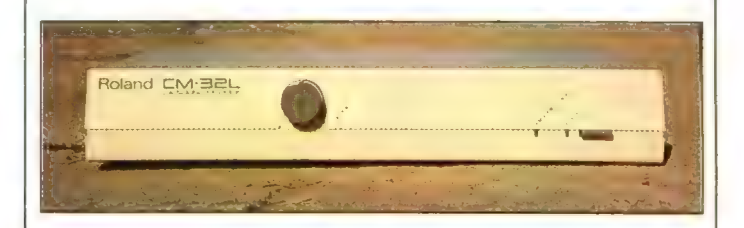




stecher ins Cabaret. Weitere Zerstreuung erhofft sich Larry in der Piano Bar des Casino's... und trifft dabei auf seine Traumfrau - Passionate Patti. Ein Date mit der Bar-Pianistin erreicht er letztendlich durch Vorzeigen der Scheidungsurkunde und Überreichen eines "lei", den er aus den in der Vulkanöffnung links von Chip'n'Dale's gepflückten Orchideen flechtet. Mit ihrem Zimmerschlüssel und einer Flasche Wein aus dem Cabaret bewaffnet fährt Larry mit dem Aufzug in die Penthouse-Suit im neunten Stock (pick floor 9) und beendet seinen Auftritt mit Einschenken des Weins. Bis dahin ist also alles beim alten: Larry sucht an diversen Orten mit den verschiedensten Frauentypen hautnah und diesmal von SIERRA gewohnte Spitzenklasse ins Spiel: Nach einer heißen Liebesnacht spricht Patti im Traum den Namen ihres Ex-Freundes aus - Larry fühlt sich betrogen und flieht endgültig vor der Zivilisation und vor den an ihr teilhabenden Menschen ins unbewohnte Innere der Insel. Wir dürfen uns nun als Patti auf die Suche nach Larry machen und beginnen mit einem typisch weiblichen Bekleidungszeremoniell. Danach geht Patti, bewaffnet mit der Flasche Wein, nochmals in die Piano Bar und nimmt den Zauberstift von der links stehenden Tafel und ihre Trinkgelder vom Klavier an sich. Bei den Umkleidekabinen füllt sie die Weinflasche und sich selbst mit Wasser. Ihr letztes Geld läßt sie bei Dale in

Das ROLAND CM-32L LA

MIDI-Sound-Modul - ein neuer Spielestandard???



Im PC-Bereich etabliert sich derzeit neben der AD-LIB-Karte die mit dem CM-32L baugleiche LAPC-1 ROLAND-Computer-Steckkarte. Vom Silpheed-Ballerspiel über King's, Police, Space und Hero's Quest bis zum neuesten SIERRA Krimi-Adventure Colonel's Bequest und der Kartenspielesammlung Hoyle's Book of Games liefert sie dem bisher soundmäßig vernachlässigten PC-User cineastischen HiFi-Stereosound.

Initiator des ganzen war SIERRA On-Line: der amerikanische Spielehersteller bereitete mit Larry II und King's Quest IV den Weg in neue Spieldimensionen vor allem durch Unterstützung des ROLAND MT-32 Sound-Moduls. Als Besitzer eines ATARI ist es Ihnen dank serienmäßig eingebauter MIDI-Schnittstelle ohne Zusatzkosten möglich, auch mit Ihren Computer-Spiele, sofern diese die Roland-Module ansprechen können, in den neuen Spielgenuß kommen. Neben SIERRA nutzen inzwischen auch die Software-Hersteller ELECTRONIC ARTS und ACTIVISION die Vielseitigkeit des CM-32L; mit weiteren Unterstützungszusagen ist zu rechnen. Das CM-32L MIDI-Sound-Modul ist eine Weiterentwicklung des ROLAND MT-32-Moduls. Die bis zu 32-stimmige, linear-arithmetische Tonerzeugung basiert weitgehend auf dem Prinzip des D-50, der ROLAND-Antwort auf die FM-synthetische Tonerzeugung des YAMAHA DX-7. Es bietet neben 128 jeden Bereich abdeckenden Instrumentalklängen und 33 digitalen PCM-Drums und -Percussions noch 33 Natursoundsamples. Von diversem Schieß- und Maschinenlärm ist über Tier- und Naturgeräusche bis hin zu Herzklopfen und Frauenlachen bzw. -schreien ein breites Spektrum abgedeckt worden. Gewisse Verschleißerscheinungen, die durch vielzählige Verwendung der Effekte bei diversen Spielen auftreten könnten, lassen sich durch geschicktes Mischen der Samples mit selbsterzeugten Klangfarben durchaus vermeiden. Somit kann diese Erweiterung in jeder Hinsicht als positiv bezeichnet werden; die Effekte passen sich hervorragend in das zeitgemäße Klangkonzept des Moduls ein. Hervorzuheben ist noch die Betriebssicherheit des Gerätes: obwohl sich bis zu 8 austauschbare Instrumente gleichzeitig in Echtzeit mischen, im Stereo-Panorama verteilen und in den Hall legen lassen, ist das CM-32L beim Testen kein einziges Mal abgestürzt.

Daß das Modul auf Program Changes und Pitchbending anspricht, ist wohl heutzutage selbstverständlich. Ungewöhnlich ist, daß auch die Natursoundsamples und Drumsounds auf MIDI-Volumen und Pitchbending reagieren.



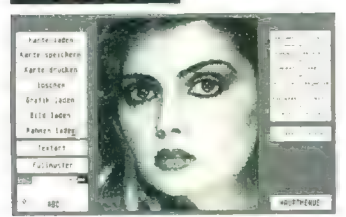
Neben den hier vorgestellten Disketten der 2000er Serie, unterhalten die PD-Pool Teilnehmer noch umfangreiche Sammlungen an PD-Paketen verschiedener Anwendungsbereiche. Aus der redaktionellen Zusammenarbeit mit dem ST-Magazin sind inzwischen 44 Disketten der 5000er Serie hervorgegangen und wer Profi-Software vor dem Kauf testen möchte, findet sein Programm in der "Freedom of Information"-Serie.

Sicher haben Sie Verständnis dafür, daß wir auf diesen Seiten nur die regelmäßigen Neuerscheinungen besprechen: Die Perlen unserer Sammlung finden Sie auf 68 Seiten der PD-Szene Fachzeitschrift. Ihr PD-Pool Anbieter schickt Ihnen PD-Szene gerne zu, denn für viele ST-Anwender ist das der Beginn einer langen Freundschaft...

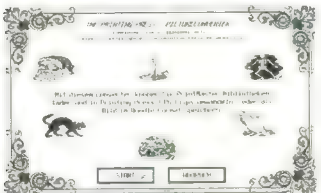
Mit freundlichen Grüßen,

Die "23"

2141



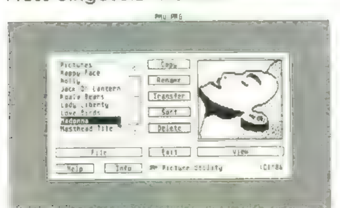
The Printing Press 3.10 arbeitet mit neuen Druckertreibern für Epson 24-Nadel, NEC Pxx und HP Laserjet. Ein nachladbarer GEM-Font sorgt nun für saubere Textgestaltung, außerdem wurden die Grußkarten- und Briefumschlagmodule überarbeitet: Absender, Anschrift, Grafiken und Zierrahmen können gespeichert und anschließend problemlos in Dokumente eingebaut werden. Das interne Grafikstudio wurde erweitert, zahlreiche Grafik-Clips und Objekte liegen auf Diskette bei. Damit erledigen Sie Gestaltungsaufgaben im Handumdrehen (s/w).



PP-Picture-Converter liest Bilddateien im SHP-Format (von Printmaster) ein und überträgt sie in das von Printing Press verwendete CPG-Format. Ein absolutes Muß für alle Printing Press Auf- und Umsteiger (s/w).

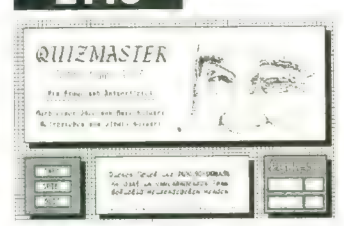
2142

Print Master Demo ist eine Fundgrube. Die darin enthaltenen Bilddateien (SHP-Format) können nach der Konvertierung mit PP-Picture-Converter in The Printing Press eingesetzt werden.



Print Master Utility ist für Print Press und Print Master gleichermaßen sinnvoll: Die beiligenden SHP-Bilddateien sind zum Einsatz in beiden Programmen geeignet. Außerdem kann mit PMU die interne Bildbank des Print Master editiert werden.

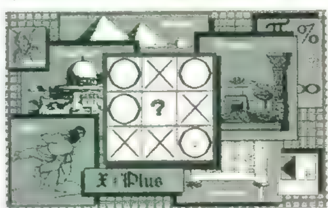
2143



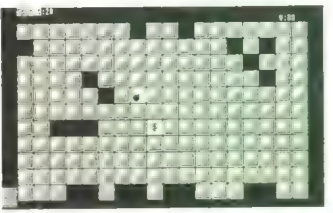
Quizmaster 0.5 ist eine Herausforderung für 1-2 Spieler. In Verlauf von drei Spielrunden wird ermittelt, welcher Spieler die an Ihn gerichteten Fragen schnell und sicher beantwortet hat (s/w).



Rennhund versetzt 1-8 Spieler in die Welt der Windhundrennen. Jeder Spieler kauft Hunde ein und läßt sie in Trainingslagern für die Rennsaison ausbilden. Preisgelder und erfolgreiche Wetten schaffen das nötige Kapital, mit dem zusätzliche Tiere auf einer der Auktionen ersteigert werden können. Ein anspruchsvolles und spannendes Strategiespiel (s/w).

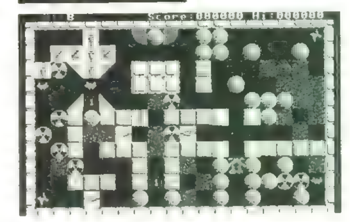


XPlus verbindet Strategieelemente des Tic-Tac-Toe mit einem aufregenden Quizspiel für 1-2 Personen. Innerhalb des Zeitlimits müssen drei in einer Linie angeordnete Fragen beantwortet werden. Der interne Editor erlaubt unendliche Variationen (s/w).

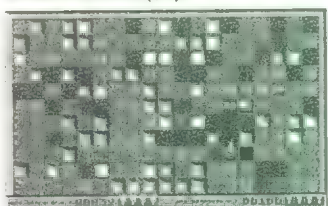


Himem besteht aus 198 kleinen Memory-Kärtchen, die von den 1-6 Spielern korrekt zugeordnet werden sollen. Das ist nicht einfach, da viele der Kärtchen einander sehr ähnlich sind (s/w).

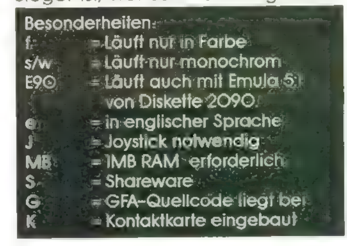
2144



Crisslefridge ist als atomares Endlager auserwählt worden. Dem Planetenbewohner Denby gefällt das überhaupt nicht: Er will die gefährlichen Abfälle beseitigen, den verseuchten Boden reinigen und die Cribblewerts an weiteren Transporten radioaktiven Materials hindern. Die attraktive Boulder-Variante ist voller Gefahren, die dem Spieler einiges abverlangen. Mit dem Level-Editor können jederzeit neue Herausforderungen erstellt werden (f, J).



Two for Two stellt zwei Panzerkommandanten auf die Probe. Sieger ist, wer sein Fahrzeug sicher durchs Labyrinth bewegt, den Hindernissen geschickt ausweicht und den Finger schnell am Abzug hat. Punkt- und Hyperspace-Felder lenken immer wieder geschickt vom Gegner ab... (f, J, J).



Besonderheiten:

f	= Läuft nur in Farbe
s/w	= Läuft nur monochrom
E90	= Läuft auch mit Emula 5 von Diskette 2090
e	= in englischer Sprache
J	= Joystick notwendig
MB	= 1MB RAM erforderlich
S	= Shareware
G	= GFA-Quellcode liegt bei
K	= Kontaktkarte eingebaut

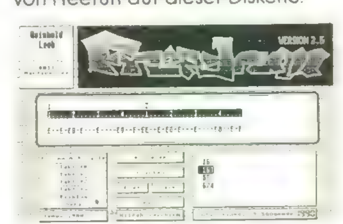
Double Bounce verwandelt den ST in einen Tennisplatz, auf dem 2 Spieler antreten (f, J, J).

Gilbert ist eine Memo-Variante, bei der innerhalb des Zeitlimits möglichst alle Spielsteine aufgenommen werden müssen. Einige Steine verdoppeln/halbieren die Punktzahl des Spielers, bzw. vertauschen sie mit der Punktzahl des Gegners (f).

Deathstar gibt Ihnen vier Leben, die Sie im Kampf um Geld und Ruhm aufs Spiel setzen können. Angriffswellen, Geldregen und starke Endmonster wollen bewältigt werden. Ansprechende Titelmusik, gute Grafik und martialische Soundeffekte zeichnen das mit STOS erstelle Spiel aus (f, J).

2145

Freedrum 2.5 bringt mit BIOLOGIC.SEQ eine neue Soundbibliothek und zahlreiche, dazu passende Songs auf den Markt. Besitzer des Runtime-Moduls erhalten außerdem die neue Version von Freerun auf dieser Diskette.



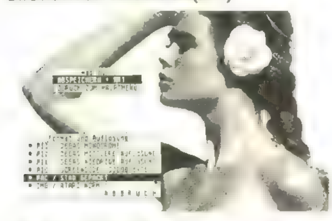
The Dark Aaah schaltet den Monitor dunkel, wenn dieser nicht benötigt wird. Der ultimative Bildschirmschoner wird einfach in den Auto-Ordner kopiert: Lautstärke aufdrehen, damit die digitalisierten Sounds voll zur Geltung kommen!

Click und **Beep** belegen die Tastatur mit digitalisierten Sounds: Trommeln Sie auf dem Zahlenblock, das ist sehr unterhaltsam.

Tatü hört man schon von weitem, plötzlich fährt ein Polizeiauto übern Monitor. Dieser Einsatz wiederholt sich von nun an alle drei Minuten.

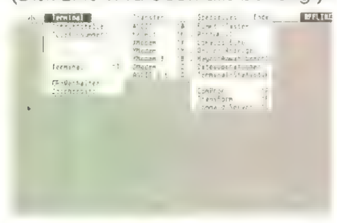
2146

Clip-Art 14 enthält 55 Bildschirme mit Männern, Frauen, Tieren, Technik und verschiedenen anderen Themen. Dia-Show und Bildkonverter anbei (s/w).



2147

Trans-Term System 1.8 von Rainer Frädrich stellt eine komplette Arbeitsumgebung für Datenreisende bereit. Wichtige Neuerung: Der ComProc muß nicht mehr programmiert werden, er lernt selbst, was er tun soll. F-Tasten im TransTerm können mit Telefonnummern belegt werden (Disk 2148 wird ebenfalls benötigt).

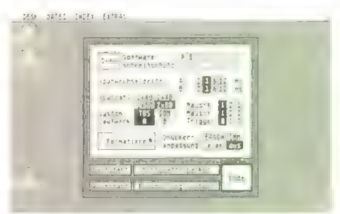


2148

Trans-Term Disk 2 arbeitet nur mit Diskette 2147 zusammen.

2149

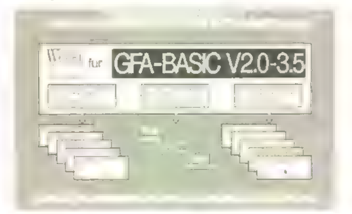
Weller HD-Tools enthält zahlreiche Programme, die den Umgang mit der Festplatte vereinfachen sollen. **Setup** lädt ACCs und AUTO-Programme, stellt Zeit und Datum, druckt Diskaufkleber. **Find File** und **Find Text** helfen bei der Suche nach bestimmten Dateien und Texten. **WT-Double** ermittelt doppelte Dateien auf der Platte. **WT-Tree** gibt Ordnerstrukturen und Pfade einer Partition auf Bildschirm oder Drucker aus. **Multi Accessory** von Henrik Alt erledigt vielseitige Aufgaben im Desktop: Umschalten der Floppy-Step-Rate für MS DOS-Betrieb, Software-Schreibschutz für Laufwerke und Partitionen, Parken der Festplatte und noch mehr...



Bootapf zaubert Apfelmännchen auf den Bildschirm während die Platte initialisiert.

2150

Weller-Tools 3.02, das Entwicklungssystem für GFA-BASIC-Programme. Der Cross-Reference-Analyser findet Fehler und macht Verbesserungsvorschläge. Weller-Tools umfaßt zahlreiche Utilities. Integrierte Hilfs- und Outline-Funktionen zur Analyse von Programmstrukturen. Handbuch mit 64 Seiten im WordPlus und ASCII-Format. Die PD-Version bearbeitet Programme mit maximal 100KB Größe, alle übrigen Funktionen entsprechen exakt der Profiversion.



Menü wurde in PD-Szene Nr. 7 vorgestellt. Das GFA-Listing erlaubt die Programmierung von Pull-Up Menüs. Die Zeitschrift PD-Szene erhalten Sie bei einem der unten angegebenen PD-Pool-Händler.

Die vorgestellten Disketten erhalten Sie exklusiv bei folgenden PD-Anbietern:

HD-Computertechnik
Pankstr. 61
1000 Berlin 65
030 / 4657028-29

OHST-Software
Nelkenstr. 2
4053 Jüchen 2
02164 / 7898

ALPHACOPY
Postfach 2161
6370 Oberursel/Ts.
06171 / 22221

=PD-Express= J. Rangnow
Ittlinger Straße 45
7519 Eppingen-Richen
07262 / 5131 (ab 17 Uhr)

Simonis Computersysteme
Ubostraße 55
8000 München 60
089 / 8632609

V.U. Volker Uecker
Hohenkamp 2
2308 Preetz
04342 / 83842

Intersoft
Nohlstr. 76
4200 Oberhausen 1
0208 / 809014

KREATIV-Software
Oberwürzbacher Str. 10
6676 Mandelbachtal
06803/3850

Duffner's PD-Center
Ritterstr. 6
7833 Endingen a.K.
07642 / 3875 od. 3739

T.S. Service
Szemere Hard&Software
Schleißheimer 127, 8 Mü 40
089 / 3089408

ST Profi-Partner
Mönkhofer Weg 126
2400 Lübeck
0451 / 505367

EU-SOFT Peter Weber
Josefstraße 11
5350 Euskirchen
02251 / 73831

Perus Computer
Weimarstr. 34
7014 Kornwestheim
07154 / 4028

LAUTERBACH-Software
Josephsplatz 3
8000 München 40
089 / 2722377

Robert Rehrl PD
Stettenerweg 8
8221 Teisendorf
08666 / 6249

T.U.M.-Soft&Hardware
Hauptstr. 67
2905 Edewecht
04405 / 6809

IDL Software
Lagerstraße 11
6100 Darmstadt 13
06151 / 58912

Weeske Computer
Potsdamer Ring 10
7150 Backnang
07191 / 1528-29 od. 60076

Peter Gerstenberg
Kafkastraße 48
8000 München 83
089 / 6377309

Schick EDV-Systeme
Hauptstraße 32a
8542 Roth
09171 / 5058-59

Händleranfragen erwünscht: Wir suchen noch PD-Anbieter und Fachhändler für gemeinsame Projekte und Veranstaltungen im PD-Bereich.

☐ Bitte senden Sie mir Heft 7 der PD-Szene. DM 2,50 liegen bei.

☐ Scheck über DM ______ liegt bei, ich erhalte die Ware verpackungs- und versandkostenfrei (Ausland: Bitte Euroscheck in der Landeswährung des Händlers)

☐ Per Nachnahme. Nur Inland! (zuzüglich DM 6,– Nachnahmegebühr).

2001	2011	2021	2031	2041		2061	2071	2081	2091	2101	2111	2121	2131	2141
2002	2012	2022	2032	2042	2052	2062	2072	2082	2092	2102	2112	2122	2132	2142
2003	2013	2023	2033	2043	2053	2063	2073	2083	2093	2103	2113	2123	2133	2143
2004	2014	2024	2034	2044	2054	2064	2074	2084	2094	2104	2114	2124	2134	2144
2005	2015	2025	2035	2045	2055	2065	2075	2085	2095	2105	2115	2125	2135	2145
2006	2016	2026	2036	2046	2056	2066	2076	2086	2096	2106	2116	2126	2136	2146
2007	2017	2027	2037	2047	2057	2067	2077	2087	2097	2107	2117	2127	2137	2147
2008	2018	2028	2038	2048	2058	2068	2078	2088	2098	2108	2118	2128	2138	2148
2009	2019	2029	2039		2059	2069	2079	2089	2099	2109	2119	2129	2139	2149
2010	2020	2030	2040		2060	2070	2080	2090	2100	2110	2120	2130	2140	2150

Diskpreis: DM 8,– *

* unverbindlich empfohlener Verkaufspreis

Lieferung an meine Adresse:

____

____

____

____

STC 9/90

Gewünschte Disketten ankreuzen und Bestellschein an einen der oben angegebenen Anbieter einsenden.

"Chip'n'Dale's", ebenso wie ihre Höschen, die sie angeheizt von dessen Show auf die Bühne wirft. Mit Schwierigkeiten, aber selbstätig, überwindet Patti das Bambusdickicht, unterwegs hält sie sich mit dem mitgenommenen Wasser am Leben. Bei der Schlucht angekommen, nimmt sie zuerst einen kräftigen Schluck Wasser, bevor sie ihre Nylon-Strümpfe um den Felsen bindet und auf diese Weise versucht, ins Tal zu gelangen. Natürlich reißen diese und sie landet auf einem Felsvorsprung inmitten des tiefen Canyon mit zwei Palmen und seltsamen Pflanzen. Die Pflanzen verwendet sie selbstverständlich zur Anfertigung eines Hanfseiles, das sie um den auf der gegenüberliegenden Seite des Canyon befindlichen Felsen wirft und danach an der Palme befestigt. Gesichert mit einem aus Kleiderfetzen gefertigten Gürtel und um zwei Kokosnüsse bereichert beginnt sie ihren Hanfseiltanz und beendet ihn mit Erfolg. Die Fortführung ihrer Suche nach Larry versucht nun ein Wildschwein zu verhindern, dem sie aber als gewitzte Frau mit einer aus BH und Nüssen gebastelten Schleuder Paroli bieten kann. Nun trennt sie nur noch ein Fluß mit reißender Strömung von ihrem Larry. Sie benutzt das im Wasser liegende Stück Holz zum weiteren Fortkommen (use log, mount log !!!) und überwindet als erfahrene Arcade-Gamerin die nächsten Hindernisse in gewieftem Zick-Zack-Kurs, bevor auch sie wie schon Larry von den Kannibalinnen gefangen genommen wird. Zusammen in einen Käfig gesperrt, klären sich alle Mißverständnisse und nach gelungenem Ausbruch dank Zauberstift findet das große Finale im SIERRA-Studio statt mit vielen bekannten Schauplätzen und Personen, unterstützt von der entsprechenden "Film"-Musik. Ein wirklich toller Schluß eines dank Patti wieder einmal hervorragenden Adventures!

Larry III wird auf Disketten geliefert. Soundunterstützung bieten die ROLAND Module MT-32 und CM-32L, Casio CZ Keyboards, ...? Als Kopierschutz fungiert das mitgelieferte "Handheft" in Form einer "Nontoonyt-Tönite"-Ausgabe, welches einerseits lediglich Fachwissen, andererseits aber auch benötigte Paß-Nummern und Seitenzahlen vermittelt, die zum weiteren Vorankommen unumgänglich sind. Die vor Spielestart zu beantwortenden Fragen entscheiden nur über den Spiellevel, wenn Sie jedoch alle fünf richtig beantwortet haben, wird bei einem "Restore" automatisch der Level geladen, unter dem Sie das Spiel abgespeichert haben. Zudem zieht sich die Postkartenschönheit ganz aus und die Jalousien bleiben beim Blick durch das Fernrohr ganz oben. Ihre Bemühungen um richtige Antworten lohnen sich also, aber auch bei keiner richtigen Antwort bleibt Ihnen der Spielspaß nicht verwehrt und es bleibt uns nur noch, Ihnen viele tolle Stunden mit diesem neuen Larry-Streich zu wünschen.

Uta Hervol

Leisure Suit Larry III

- Adventure
- Hersteller: SIERRA On-Line
- englisches Handbuch, gleichzeitig Paßwort-Abfrage
- Maus, Joystick, Tastatur
- Soundunterstützung: ROLAND MT-32, CM-32L-Sound-Modul

Ohne tiefer ins MIDI-SysEx-Datendickicht einzusteigen, können Sie über die inzwischen standardisierten Controller-Nummern Volumen der einzelnen Parts (über Controller-Nr. 7 und über Expression-Pedal-Regler (Controller-Nr. 11) für Sub-Volume) Balance (Controller-Nr. 10), Modulationsrad, meist für Vibrato verwendet (Controller-Nr. 1) und Sustain-Pedal (Controller-Nr. 64) verändern. Leider sind Mastervolumen- und Hallveränderungen sowie einige weitere Parametereinstellungen, z.B. Pitchbend-Range, nur über zum CM-32L kompatible MT-32-Sound-Editoren oder auf dem Wege der Selbstprogrammierung möglich.

Auf der Rückseite des im ROLAND-DG Grau gestylten Gerätes befinden sich jeweils eine MIDI/In, /Out und /Thru-Buchse, zwei Stereoausgänge für Verstärker (6,3 mm Mono-Klinke), ein Kopfhöreranschluß (6,3 mm Stereo-Klinke) sowie eine 9-Volt Netzadapter-Buchse. Die spartanische Frontplatten-Ausstattung besteht lediglich aus einem Volumen-Regler, dem Netzschalter mit zugehöriger LED-Anzeige und einem MIDI-Message Indikator.

Zum Lieferumfang gehören 1 MIDI-Kabel, 2 Audio-Kabel mit Cinch-Klinke-Adapter, ein User-Manual mit den notwendigsten Erklärungen inklusive MIDI-Implementation und ein kleines MIDI-Guide Book.

Mit diesen Features ist das CM-32L eine ideale Ergänzung zu Ihrem ST, weil für midifizierte Musik als "All In One"-Workstation und Spieleanwendung gleichermaßen geeignet. Somit ist der auf den ersten Blick doch hoch erscheinende Preis durchaus als günstig zu bezeichnen, bekommt der einsteigende MIDIaner doch alles, was sein Herz begehrt, kompakt in einem ansprechend verpackten Modulgeliefert. Für Profis und Fortgeschrittene dürfte das CM-32L in erster Linie als Kompositionswerkzeug interessant sein - die Technik des Moduls ist für Studio-Einsatz nicht mehr ganz "State of the Art". Aber wenn Sie dann noch Spiele-Freak sind, ist der Erwerb des CM-32L schon fast ein Muß.

Gerhard Schedel

ROLAND CM-32L LA MIDI-Sound-Modul

(MagicMusic, Öhringen, knapp DM 1.300,-)

Verbesserte Version des bekannten ROLAND MT-32-Sound-Moduls 32-stimmiges Sound-Modul, 128 Klangfarben, davon 8 gleichzeitig spielbar, zusätzlich 33 digitale Drumsounds, 33 Effektklänge, 4 verschiedene Digitalhall-Arten

Stereoausgang für Verstärker (6,3 mm Klinke, mono) und Kopfhörer (6,3 mm Stereo-Klinke) Volumeregler, MIDI Message Indicator, Power Indicator

MIDI-In, -Out, -Thru

Im Lieferumfang enthalten:

1 MIDI-Kabel,
2 Audio-Kabel mit Cinch-Klinke-Adapter,
Owner's-Manual

Spezial-Software :

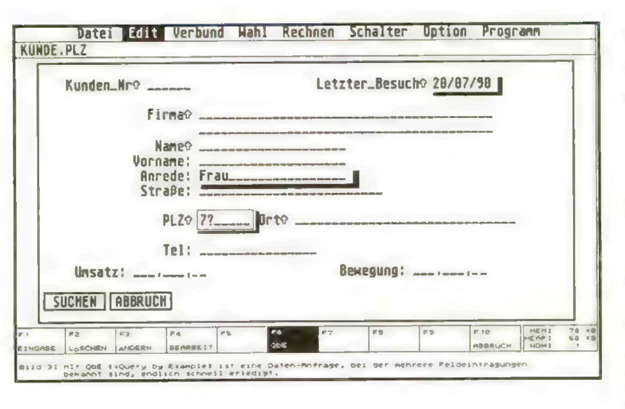
alle neueren SIERRA-Adventures, z.B. Larry II und III, Space Quest, Colonel's Bequest neue Electronic Arts-, Activison-Games
Alle MT-32-Editoren diverser Hersteller



Adimens ST plus

Version 3.1

In Heft 11/89 durften wir ein stark erweitertes Adimens vorstellen. Die damalige Frage, ob das "PLUS" im Produktnamen berechtigt sei (wurde dort nicht untersucht), scheint sich in positiver Richtung zu beantworten. Adimens ST plus entstand aus dem ersten professionell angewandten Datenbankprogramm für den ATARI ST. Seit diesen Anfangstagen wurde es stetig weiterentwickelt und verbessert.



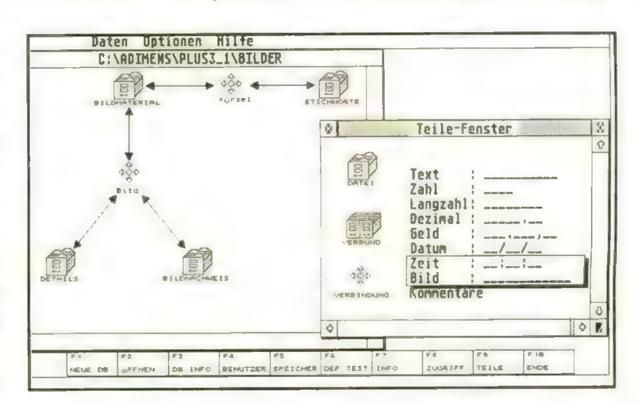
Pünktlich zur ATARI-Messe in Düsseldorf liegt eine weitere Version vor, die Adimens schon wieder einmal einen Sprung nach vorn machen läßt. Vornehmlich zwei Neuerungen sind es, die in der Version 3.1 auffallen.

Bilder in der Datenbank

Nicht nur Bildjournalisten, Illustratoren oder DTP-Grafiker stehen vor dem Problem, Text- und Bildvorlagen in sinnvollem Zusammenhang zu archivieren. Sie verlieren schnell die Übersicht. Auch technische Zeichner und Konstrukteure finden sich oft nicht mehr in der Vielzahl der katalogisierten Einzelteile zurecht. Und es kommt ein weiterer Aspekt hinzu: Erfahrungsgemäß werden im Bereich der technischen Konstruktion lieber zu 80% Teilezeichnungen neu erstellt, als auf ähnliche zurückzugreifen. Auch die Vielfalt der Teilenummern, Produktfamilien und Lieferanten wird unüberschaubar.

Mit Adimens fließt nun die Verwaltung von Text und Bildern zusammen. Vorbei ist die Zeit der getrennten Archivierung und Katalogisierung. Im Vereinbarungsteil INIT wird in der Maske der Datentyp "Bild" festgelegt. Dort soll später der Name einer externen (also getrennt abgespeicherten) Bilddatei eingetragen werden. Beim Aufruf der Maske im Ausführungsteil EXEC erscheint ein Bild-Kästchen, das beim Anklicken jene Bilddatei lädt, das unter dem angegebenen Namen vorher abgespeichert war. Durch erneutes Klicken auf BILD kann ein anderes Bild angesehen und eingetragen werden. Diese Lösung verhindert z.B. durch Vereinbarung als eindeutigen Schlüssel eine unerwünschte Mehrfachverwendung.

Lern- und Wissensdatenbanken lassen sich nun anreichern um Bilder, z.B. Lösungen mathem. Aufgaben oder Veranschaulichungen, und unter Ausnutzung der Möglichkeit des Verzweigens zwischen einzelnen Teilgebieten (Dateien) können Sie auch komplexe Bestände verwalten.



Zwischen Suche und Wahl

Es ist allgemein üblich, in einer Adreßkartei nicht nur nach einem Kriterium suchen zu lassen, sondern gleich nach mehreren. Die einfache Suchfunktion in Adimens reicht dafür nicht aus. Auch in der Wahldefinition war das Ziel erst über lange Beschreibungen über mehrere (Such-)Schlüssel möglich.

Mit der neuen Funktion "QbE" (Query by Example = Abfrage durch Vorgeben einer Musterbedingung) müssen lediglich die bekannten Angaben in die betroffenen Eintragsfelder eingegeben werden, um nach Bestätigen der Maske sofort einen Suchvorgang auszulösen.

Da erfahrungsgemäß der überwiegende Teil der Anfragen an eine Datenbank in die Kategorie der "gleichbedingten UND-Verknüpfungen" fällt, profitieren insbesondere Adimens-Anwender im kommerziellen Bereich von dieser in der Praxis wesentlichen Verkürzung des Suchvorgangs.

Up-to-Grade

Ein Upgrade ist sowohl über ausgewählte Adimens-Stützpunkthändler möglich als auch direkt bei der ADI. Wird Adimens ST plus 3.1 selbst abgeholt, zahlen registrierte Adimens ST plus 3.0-Anwender lediglich DM 50,—, bei Zusendung durch die ADI DM 80,-. Gegen die Originaldisketten und Angabe der Registriernummer erhält der Kunde dann die neuen Originaldisketten plus Zusatzdokumentation für sein bestehendes Adimens ST plus 3.0-Paket.

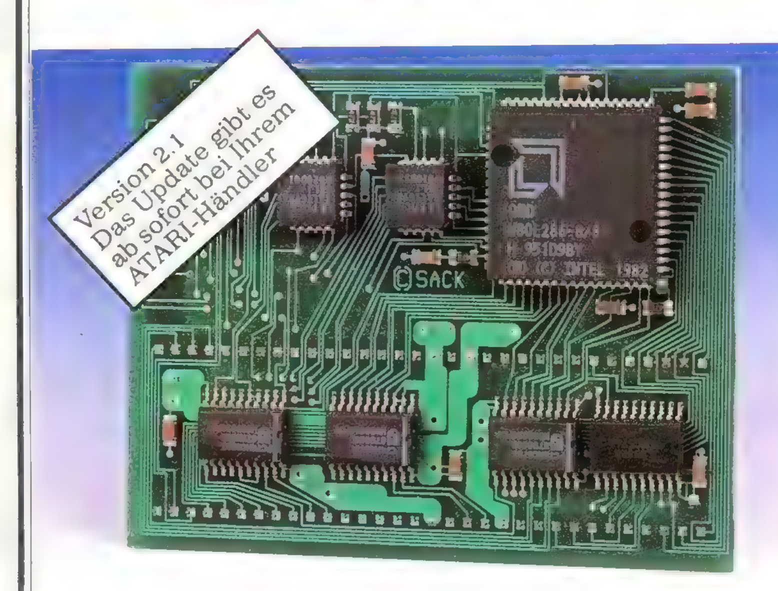
Adimens 2.x-Anwender erhalten gegen ihre Originaldisketten für DM 150,- ein komplett neues Paket Adimens ST plus 3.1 und können sich zu dritt zu einer Sammelbestellung zusammentun (DM 300,-). Der empfohlene Ladenverkaufspreis für Adimens ST plus 3.1 beträgt DM 399,-.

DK

Bezugsadresse:

ADI Software GmbH
Hardeckstr. 5
7500 Karlsruhe 21
Tel.: 0721/570000

Neu! Das UPDATE V 2.0



Die Pluspunkte:

+ Norton Faktor 6.7
+ Hohe Kompatibilität
+ Accessory
  schneller Wechsel
  zwischen TOS und MS-DOS

NEU AB VERSION 2.1:

+ ATARI Laserdrucker
  ist an AT-Speed angepaßt
+ Windows 3.0
  läuft im Protected Mode

AT Speed – der schnelle MS-DOS-Emulator (PC/AT) – Norton Faktor 6.7 – durch schnellen 80286 Prozessor:

- verwaltet unter MS-DOS 24 Partitionen mit bis zu je 32 MB
- bootfähig von internem, externem Laufwerk oder Festplatte
- unterstützt internes Laufwerk 360/720 KB
- unterstützt externes Laufwerk 3,5"/5,25", 360 KB und/oder 720 KB
- unterstützt 1,4 MB-Disketten-Laufwerkslösung von Digital-Image und Maxon-Computer
- unterstützt die serielle und parallele Schnittstelle
- unterstützt alle bekannten Fest- und Wechselplatten, auch die ct'-Festplattenlösung (mit OMTI-CONTROLLER)
- enthält die Eigenschaften des AT's
- 80286-Prozessor verwaltet den gesamten RAM-Speicher des ATARI, (Treiber-Programme wie z. B. Himem.sys sind lauffähig)
- 704 KB DOS-Speicher, Max. 3 MB Extended /Expanded Memory
- ATARI-Maus als Microsoft-kompatible Maus nutzbar
- Sound wird völlig unterstützt
- AT Speed wird dem 68.000er-Prozessor aufgelötet, *die saubere Lösung wie bei PC-Speed*
- beim MEGA ST über Speed-Bridge aufsteckbar
- unterstützt Grafikkarten: CGA, TOSHIBA, HERCULES, ATT, OLIVETTI, TANDY (16 Farben)
- hat hohe Kompabilität des PC Speed
- bietet die Möglichkeiten des PC Speed und mehr...
- kleine, handliche Platine in SMT (Surface Mounted Technology)
- und viele andere Vorteile

Vertrieb weltweit:

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

In Deutschland: Über 400 ATARI-Händler informieren, beraten, bauen ein, betreuen

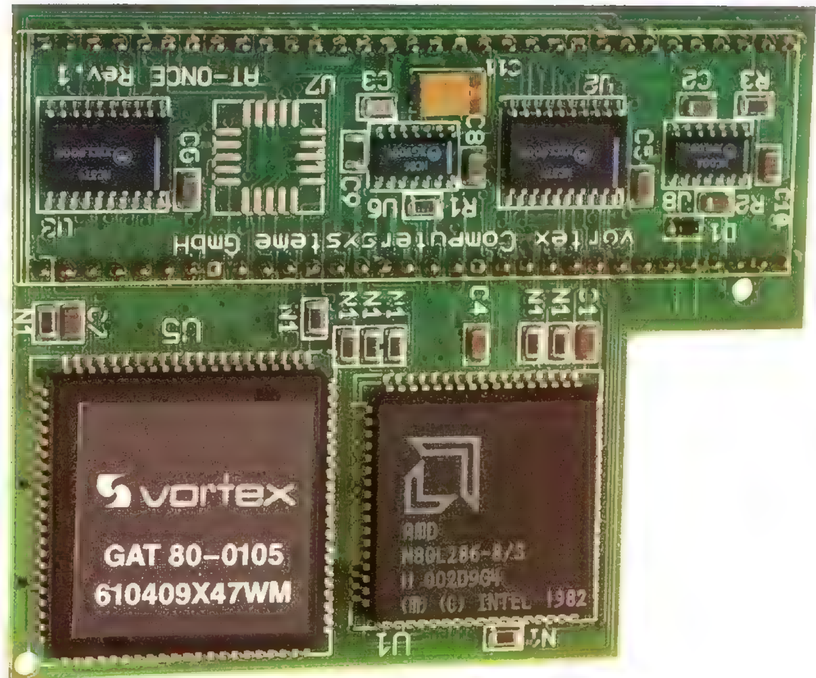
Österreich: Darius
Inh. K. Hebein
Hartlebengasse 1-17/55
A-1220 Wien
Tel.: 00 43-222-23 95 80

Schweiz: Data Trade AG
Landstr. 1
CH - 5415 Rieden-Baden

Für alle anderen Länder:
COMPO SOFTWARE GmbH
Postfach 1051
D-5540 Prüm (FRG)
Tel.: 0 65 51 / 62 66

Händler:

Rufen Sie an.
Wir nennen Ihnen gern Ihren Händler



ATonce

Der ST wird zum ATARI

Atari hat es versäumt, dem ST einen vernünftigen Standard mit auf den Weg zu geben. Obwohl sich der ST zu einem eigenen Standard entwickelt hat, wollen immer mehr ST-Anwender dem "Industriestandard" nacheifern - schneller, weiter, höher.

Die bisherigen PC-Emulatoren für den ST setzten auf den 8086-, 8088- oder V30-Prozessor von Intel bzw. NEC. Daß diese Prozessoren inzwischen vom ebenfalls veralteten 80286 bzw. 80386 abgelöst wurden, hat sich herumgesprochen. Wieso sollte man also nicht auch einen Emulator anbieten, der mit einem solchen Prozessor arbeitet?

Vortex setzte diese Frage in die Tat um und stellte zur CeBIT '90 den ATonce vor - den ersten AT-Emulator für den ST. Einige Zeit später war er tatsächlich erhältlich und traf auch in der Redaktion ein. Einem ATonce-Paket entnehmen wir eine Platine, ein Handbuch und eine Diskette mit der Installations- und Betriebs-Software und einen gedrehten Präzisionssokkel. Die Platine ist sauber aufgebaut. Auf ihr befinden sich ein 80L286-Prozessor (voll kompatibel zum 80286), ein Gate-Array und vier SMD-Chips sowie zwei Pfostenreihen zum Einbau.

Der Einbau

Wer bereits einen PC-Speed eingebaut hat, kennt die Installationsprozedur. Der ATonce muß auf den 68000-Prozessor aufgelötet werden. Wer das nicht mag und einen Mega ST besitzt, kann auch die lötfreie "SpeedBridge" benutzen. Aber auch STE-Besitzer können aufatmen, denn Vortex bietet (gegen Aufpreis) einen Steckadapter an, der den Betrieb des ATonce im STE möglich macht.

Auf knapp 30 Seiten wird im DIN A6-großen Handbuch überaus ausführlich das Auspacken und Einbauen des Emulators beschrieben - damit dürften selbst Laien keine Probleme haben. Leider sind jedoch die Installation und Inbetriebnahme auf lediglich sechs Seiten beschrieben und sehr dürftig. Das Installationsprogramm ist jedoch recht komfortabel und schwächt das Manko ein wenig ab. In das Handbuch sollte Vortex allerdings noch etwas Zeit investitieren.

ATonce unterstützt problemlos den SLM804-Laserdrucker. Dabei ist auswählbar, ob man den Diablo-Treiber von Atari oder den Laserbrain-Treiber von DMC installieren will. Weiterhin befinden sich auf der nicht kopiergeschützten Systemdiskette diverse Treiber, Beispieldateien, Programme zur Einstellung der Grafikemulation und zum Zurückkehren in den ST-Modus.

Die Installation...

...ist recht einfach zu bewerkstelligen. Hier sind Diskettenlaufwerke (1 oder 2, 3,5 oder 5,25 Zoll, 40 oder 80 Tracks, Laufwerk A oder B) und Festplatten auszuwählen. Alle ATARI-Partitionen können auch von ATonce aus genutzt werden, bis zu 24 Stück (C-Z) können installiert werden. Damit ist ATonce voll kompatibel zu MS-DOS. Eine Partition darf höchstens eine Größe von 32 MB besitzen. Das ist zwar nicht ganz MS-DOS-konform, aber noch erträglich.

Auch bei den Grafikemulationen darf man seine Auswahl treffen. Neben CGA- und Hercules-Modus können auch der Toshiba T3100- oder Olivetti-Modus gewählt werden. Ist zusätzlich auch OverScan im Rechner installiert, ist ATonce sogar in der Lage, die gesamte Hercules-Auflösung (720x348) auf einmal anzuzeigen. Auf einem Farbmonitor läßt sich allerdings nur die CGA-Auflösung benutzen. Booten ist entweder von Diskette oder von Festplatte möglich. Das Festplattenformat ist kein eigenes (wie beim SuperCharger), es kann also auch von der Atari-Seite auf die Partition zugegriffen werden, ohne einen kompletten Absturz zu riskieren. Letztlich läßt sich im Installationsprogramm noch einstellen, wie der vorhandene Speicher aufgeteilt werden soll. Die bei einem Mega 4 verbleibenden 3 MB lassen sich dem Expanded- oder Extended-Memory zuweisen.

Durchstarten

Nach dem Start des eigentlichen Emulators glaubt man wirklich, einen AT vor sich zu haben - der Speichertest deutet darauf hin. Nach dem Booten stehen bis zu zwei Druckerschnittstellen zur Verfügung

(der Laser liegt auf LPT1:, alle anderen dann auf LPT2:), auf jeden Fall jedoch zwei serielle Schnittstellen. Auf COM1 befindet sich die Atari-Maus, die hier als Microsoft-kompatible benutzt werden kann, an der anderen läßt sich ein beliebiges anderes serielles Gerät anschließen. Vortex liefert leider den Maustreiber nicht mit, so daß man ihn sich anderweitig besorgen muß. Der Treiber ist allerdings ohne Maus nicht erhältlich - fragwürdig ist also, wo er beschafft werden kann.

ATonce ist merklich schneller als PC-Speed und SuperCharger. So beträgt der Norton-Faktor denn auch 6,7. Die theoretische Geschwindigkeitssteigerung von 1,5 zum SuperCharger wird praktisch allerdings nicht erreicht. Trotzdem ist es wesentlich komfortabler, mit der schnelleren Geschwindigkeit zu arbeiten. Die anderen Meßwerte entnehmen Sie bitte der Tabelle.

Mit Programmen hat der ATonce kein Problem. GEM, Wordstar, Word, Lotus, dBASE und Windows 3.0 funktionieren einwandfrei. MS-DOS wurde in der Version 3.3 und 4.01 getestet und bereitet keinerlei Schwierigkeiten. Der ATonce weiß auch, wie er mit Extended und Expanded Memory umzugehen hat. Das einzige kritische Programm, das wir gefunden haben, ist Telix, das auf dem ATonce nur empfangen kann. So muß der DFÜler hier auf Procomm zurückgreifen.

Fazit

Der ATonce ist ein guter AT-Emulator. Wenn man bedenkt, daß Vortex solcherlei Geräte niemals vorher gebaut hat, ist es überraschend, wie betriebssicher die erste Version des Emulators funktioniert. Für DM 498,- ist derzeit kein anderer AT-Emulator erhältlich - damit bietet der ATonce ein sehr gutes Preis/Leistungsverhältnis. Kein anderer AT-Emulator bietet zur Zeit eine solches Angebotsspektrum.

MP

Bezugsadresse:

vortex Computersysteme GmbH
Falterstr. 51-53
7101 Flein bei Heilbronn
Tel.: 07131/5088-0

Das Benchmark-Programm "PTIME"

Zuerst wurde ein Programm gestartet, das einige Prozessorbefehle hintereinander ausführte und danach die Zeit maß. Es wurden gemessen: Loop; Inc; Dec; Mov Const -> Reg; Mov Byte -> Reg; Mov Word -> Reg; Add; Sub; Imul Word; Idiv Word; Not -> Reg; Ror -> Reg; Cmp; Push/Pop; Call; Call -> Parameter; Int/Iret; Shr (diese Tests jeweils ....mal); 1 MB Memory Move; 20 s. Timing Loop; Single Precision Test; Double Precision Test (jeweils einmal). Die Zeit, die die Emulatoren jeweils insgesamt benötigt haben, finden Sie in der Tabelle unter "Benchmark". Die Ergebnisse dieses Tests spiegeln vor allem wider, wie schnell der Prozessor die Befehle abarbeitet.

Die Timer-Taktung

Im dritten Test wurde getestet, ob alle Timer richtig getaktet werden. Dazu wurde mit einem einschlägigen Programm die angebliche Taktfrequenz des Prozessors gemessen. Die Ergebnisse finden Sie in der Tabelle. Die Werte sind natürlich keine richtigen CPU-Geschwindigkeiten, sondern sollen nur verdeutlichen, was ein "normales" Meßprogramm herausfindet (*).

Der Wordstar-Test

Wordstar ist eines der am häufigsten benutzten Textverarbeitungsprogramme für PC- und AT-kompatible Computer. Deshalb starteten wir auch mit diesem Programm einen Test. In einem 48 kB langen Text tauschten wir 5180mal ein "e" gegen ein "a" aus. Dabei wurden alle Austauschungen auf dem Monitor angezeigt. Zum Vergleich: Tempus braucht für die gleiche Aktion 2,85 Sekunden (und ist damit über 20mal schneller). Wordplus bezwingt Wordstar mit 43,75 Sekunden (1,29mal so schnell).

Der Norton-Faktor

Der Norton-Faktor ist der Standard-Test, um die Geschwindigkeit eines MS-DOS-Rechners feststellen zu können. Mit ihm wird eindeutig gezeigt, wie schnell ein PC (bzw. in diesem Fall ein Emulator) im Vergleich zu einem original-IBM-XT ist. Der IBM-XT würde beim Norton-Faktor ein Ergebnis von 1,0 liefern.

Die Tests

Um die Emulatoren miteinander vergleichen zu können, unterzogen wir sie einigen harten Tests. Mit dem Benchmark-Programm "PTIME" stellten wir fest, wie lange die Prozessoren für die Abarbeitung der Befehle brauchen. Schließlich kann man keinen schnell getakteten Prozessor gebrauchen, wenn die Befehle nur langsam weitergegeben werden.

Der nächste Test sollte zeigen, inwieweit man mit einem normalen Geschwindigkeitsmeßprogramm die Taktfrequenz des Prozessors feststellen kann. Dazu müssen nämlich alle Timer exakt nachprogrammiert sein. Das überraschende Ergebnis: Die wirkliche Taktfrequenz hat das Programm nicht herausgefunden. Erstaunlich ist auch, daß bei diesem Test der SuperCharger nur unwesentlich langsamer ist als der ATonce. Die genauen Ergebnisse entnehmen Sie bitte der Tabelle.

Der letzte Test war ein Praxis-Test. In einem 48 kB langen Text tauschten wir jeweils "e" gegen "a" aus. Die Austauschstellen wurden jeweils angezeigt. Dadurch erhalten Sie auch einen realitätsnahen Praxiswert, den kein Norton-Faktor oder Landmark-Test angeben kann!

MP

	Benchmark (s)	Disk-Transfer(bps)	CPU (MHz)*	Wordstar	Norton-Faktor
ATonce	80.427	133086	15,00*	55,75*	6,7
SuperCharger	91,491	95254	13,58*	61,00	4,0

Tabelle 1: Die einzelnen Testergebnisse auf einen Blick



Ein Wort in eigener Sache

In den Jahren, die unsere Zeitschrift existiert, haben wir immer wieder versucht, durch die Beantwortung der bei uns eingehenden Briefe ein wenig Licht in das Dunkel zu bringen, das bei der Arbeit mit dem ATARI ST schon so manch einen aus der Fassung bringen konnte - eine Tatsache, die nicht nur Ihnen, verehrter Leser, sondern auch uns oft genug zu schaffen machte. Nichtsdestotrotz haben wir uns bemüht, die Probleme zu lösen und diverse Leserbriefe zu veröffentlichen, da wir der Meinung waren, daß die jeweilige Thematik auch einen größeren Leserkreis interessieren könnte. Trotzdem gibt es immer wieder Briefe, die wir nicht beantworten können oder dürfen. Damit Sie nicht allzusehr enttäuscht zu sein brauchen oder keine Antwort erhalten, möchten wir Sie bitten, sich an folgende Spielregeln zu halten, die sich aus unserer Erfahrung ergeben haben. Fällt Ihr Brief nicht unter die folgenden Kriterien, hat er gute Chancen, positiv beantwortet oder wenigstens als Hilferuf an unsere Leserschaft gedruckt zu werden.

1. Leider gehen immer wieder Briefe mit dem Wunsch ein, ein Produkt für diesen oder jenen Anwendungsfall vorzuschlagen, verschiedene Produkte bezüglich der Vor- und Nachteile gegeneinander abzuwägen und zu bewerten. Es ist uns aus Wettbewerbsgründen nicht erlaubt, ein bestimmtes Produkt zu favorisieren, selbst wenn wir das eine oder andere in der Redaktion überzeugt einsetzen. Wir können Sie in diesem Fall ausschließlich auf die von uns möglichst objektiven Tests und eventuell anstehende Fachmessen hinweisen. Bedenken Sie bitte, daß auch wir nicht jede Textverarbeitung, jedes Malprogramm und so weiter kennen und bestimmte Produkte dadurch in das Abseits drängen würden.

2. Oft erreichen uns Briefe, die sich positiv oder auch negativ über bestimmte Händler, Softwarehäuser oder deren Produkte auslassen. Sicherlich interessieren uns solche Bemerkungen. Bitte haben Sie aber Verständnis, daß wir weder Lob noch Tadel abdrucken dürfen, da diese Aussagen meist subjektiv sind. Anders sieht die Sache beispielsweise bei Gerichtsurteilen aus, die Sie, verehrte(r) Leser(in), erfochten haben.

3. Aufgrund der Vielzahl an Briefen, die uns täglich erreichen, sind wir leider nicht in der Lage, Programmfehler anhand von Listings oder ähnlichem zu korrigieren. Dennoch sollte ein Problem möglichst detailliert beschrieben sein, denn Ferndiagnosen sind prinzipiell sehr schwer, jedoch mit genauerer Angabe der Symptome eventuell durchführbar.

4. Von Zeit zu Zeit erreichen uns Briefe mit der Bitte, die Adresse des Lesers zwecks allgemeiner Kontaktaufnahme zu veröffentlichen. Würden wir dies in die Tat umsetzen, würde sich der Umfang des anderen redaktionellen Teils beträchtlich verkleinern. Ausnahmen stellen Leser in fernen Ländern dar, für die eine Kontaktaufnahme im eigenen Land recht schwierig ist.

Zum Schluß sollen ein paar Tips eventuell voreilig geschriebene Briefe verhindern.

1. Wenn Sie ein Problem bezüglich einer bestimmten Problematik haben oder an einem bestimmten Produkt interessiert sind, finden Sie interessante Artikel darüber eventuell in vorhergehenden Ausgaben userer Zeitschrift. Zur Auswahl eignet sich das Jahresinhaltsverzeichnis besonders gut, das immer am Jahresende in der ST Computer abgedruckt wird.

2. Sollten die Probleme mit der Handhabung eines Produktes zu tun haben, wenden Sie sich zunächst an Ihren Händler und über diesen an den Distributor beziehungsweise an das Software-Haus. Die Wahrscheinlichkeit, daß Ihnen das Software-Haus weiterhelfen kann, ist um ein Vielfaches höher als die, daß wir Ihnen helfen können.

3. Lesen Sie aufmerksam die Leserbrief-Seite. Viele Fragen wiederholen sich immer wieder, obwohl wir bestimmte Probleme schon mehrfach angesprochen haben.

Antwort auf störrischen Drucker

Das in Heft 6/90 von Herrn Schloßer angesperochene Problem dürfte typisch für Drukker aus der "Vor-NLQ-Zeit" sein. Signum! ist so schön (und langsam), weil es den Drukkerkopf vertikal um jeweils 1/216 inch bewegt (1/3 Nadelabstand). Der Atari SMM-804 interpretiert diese Steuersequenz als 1/144-inch-Vorschub. Zu kleineren Schritten ist er leider nicht in der Lage. Nach dem zweiten Vorschub druckt die erste Nadel also über die erste Reihe der zweiten Nadel. Der dritte Vorschub sollte dann 20/216 inch betragen (oder 23, wenn Signum! alle 9 Nadeln einsetzt). Hier entsteht nun die häßliche Lükke. Herr Schloßer sollte also die Suche nach dem DIP-Schalter abbrechen. Vielleicht kann er ja ein kleineres Zahnrad auf den Walzenmotor setzen?

Hubert Achthaler, 5100 Aachen

*

Probleme mit Dialogboxen

Seit einiger Zeit beschäftige ich mich mit der GEM-Programmierung. Dabei benutze ich das RCS 1.4 von Digital Research und Omikron .BASIC 3.0. Programmiere ich Dialogboxen mit edierbaren Eingabefeldern, in denen der Cursor nach einem FORM_DO links am Textanfang stehen soll, nimmt das Programm keine Eingaben an und der Cursor läßt sich nicht innerhalb der Textzeile bewegen. In entsprechender Literatur ist zu lesen, daß in der Komponente PTEXT das Zeichen "@" bei der Dialogverwaltung den Cursor an den Zeilenanfang springen läßt. Dies geschieht auch, doch Eingaben werden nicht akzeptiert. Was mache ich falsch? Ist unter Omikron.BASIC 3.0 eine derartige Dialogverwaltung nicht möglich? Ich wäre sehr froh, wenn Sie mir diesbezüglich weiterhelfen könnten.

Jörg Friese, 3000 Hannover 91

Red.: Natürlich ist eine solche Dialogverwaltung auch in Omikron.BASIC möglich. Allerdings muß vor einem FORM_DO das Eingabefeld gelöscht werden. Das läßt sich leider aber nicht direkt im RCS durchführen. Hier muß nach wie vor in PTEXT etwas angegeben werden. Es ist allerdings ganz leicht mit einem mehrfachen Zeigergehangel im eigenen Programm zu bewerkstelligen. Am besten schreibt man sich dafür eine Prozedur. Dazu gehen Sie folgendermaßen vor:

Die Adresse des Objektbaums müßten Sie schon mit RSRC_GADDR erhalten haben. Damit errechnen Sie dann die Objektadresse Ihres Textfeldes (Objektadresse= Baumadresse+ 24*Objektindex). Nun benötigen Sie die Adresse der TEDINFO Ihres Objektes [LPEEK(Objektadresse+12)], aus der Sie wiederum die Adresse von PTEXT Ihres Textfeldes erhalten [LPEEK (Tedinfoadresse+0)]. Nun sind wir endlich soweit. In diese letzte Adresse schreiben Sie eine Null, und das Textfeld ist gelöscht und kann neu beschrieben werden. Das müßte problemlos in Omikron .BASIC umzusetzen sein.

*

FAX mit dem ST?

Ich besitze einen Atari 1040 STE sowie einen Citizen Swift 2-Drucker. Wie ich gehört habe, soll es eine Möglichkeit geben, diese Anlage auch zum Senden von FAXen zu verwenden. Ich wäre Ihnen sehr dankbar, wenn Sie mir in dieser Angelegenheit weiterhelfen könnten.

Karlheinz Geiser, 6700 Ludwigshafen

Red.: Uns ist bislang kein Programm zum Versenden von FAXen bekannt. Allerdings kämen die Kosten eines solchen Systems weit über das eines normalen FAX-Geräts: Scanner, FAX-Modem, Festplatte, Software, Rechner und Monitor würden weit über 5000,- DM kosten und nur einen Bruchteil der Möglichkeiten bieten, zumal die Übertragung einer Seite knapp viermal so lange dauernd würde! Sollte in der Leserschaft trotzdem ein FAX-Programm für den ST mit SendFAX-Modem bekannt sein: Bitte melden!

*

Zu defektem Monitor

In meinem Schreiben beziehe ich mich auf den Leserbrief von Georg-Friedrich Choitz aus Hannover aus der Ausgabe 6/90. Er schilderte dort ein Monitorproblem, das so auch vor zwei Jahren bei mir auftrat.

Im Monitor SM124 befindet sich u.a. das IC TDA 1170 (IC 601), welches für die Vertikalablenkung zuständig ist. Dieses beinhaltet einen Oszillator, der durch externe Beschaltung gesteuert wird. Hier ist es im wesentlichen der Vertica-Hold- (V.Hold-)Trimmer und ein 100 nF-Kondensator (C 603). Dieser Kondensator kann bei bestimmten Temperaturen und Signalen durchschlagen bzw. ausfallen. Dadurch gerät die Vertikalablenkung durcheinander. Um sicherzugehen, daß der Fehler nicht am Computer selbst liegt, sollte man das V.Sync-Signal an der Steckverbindung Monitorplatine-Monitorkabel (P 301) mit einem Oszilloskop kontrollieren. Dieses Signal muß bei korrekter Triggerung stillstehen. Die Angaben in Klammern entsprechen dem Aufdruck auf der Platine älterer SM124. Je nach Baujahr des Monitors kann es wegen Platzproblemen auch schwierig sein, den Kondensator auszutauschen. Mit dieser Bezeichnung kommt ein Hobby-Elektroniker hoffentlich zurecht. Ich warne trotzdem vor den evtl. lebensgefährlichen Hochspannungen bei Bildschirmgeräten!

Thomas Steiner, 5090 Leverkusen

Unmögliche Fileselectbox

Seit April '89 bin ich Besitzer eines Mega 1. Seit einiger Zeit habe ich Probleme mit der vom Betriebssystem bereitgestellten Fileselectbox, die ich des öfteren durch die MAXON-Box ersetzt habe. Zur Beschreibung des Fehlers, der beim Laufwerkswechsel von A: nach B: auftritt, füge ich die folgenden Bilder an. Es erscheint eine übliche Alertbox, die zum Diskettenwechsel bei der Arbeit mit nur einem Laufwerk aufruft (Bild 1). Was dann passiert, zeigt das nächste Bild: Der Cursor befindet sich, will man diesen seltsamen Pfadnamen ändern, außerhalb der Box. Betätigt man die Backspace-Taste, ist er so zu sehen (Bild 2). Was kann ich tun? Gibt's das öfter? Hilft hier TOS 1.4? Ich möchte einfügen, daß die Box trotzdem ihren Dienst verrichtet. Allerdings verweigert sie ihn auch des öfteren: Der Default-String blinkt munter vor sich hin, der Cursor wandert über den Schirm, eine Anwahl von Buttons oder gar die Eingabe in die String-Felder ist nicht mehr möglich; kurz: Der Rechner hängt sich auf.

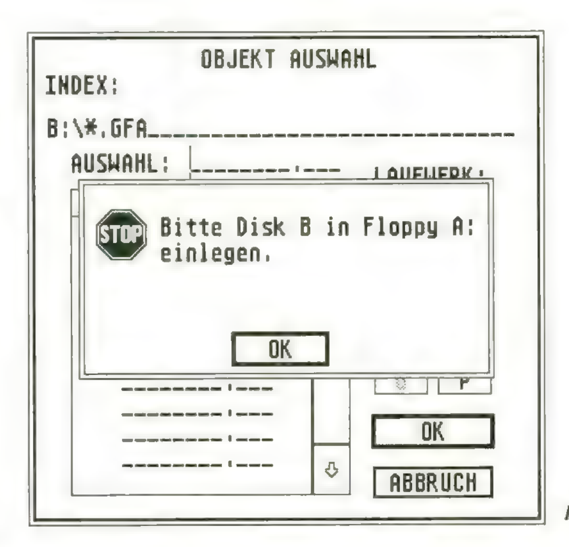


Bild 1

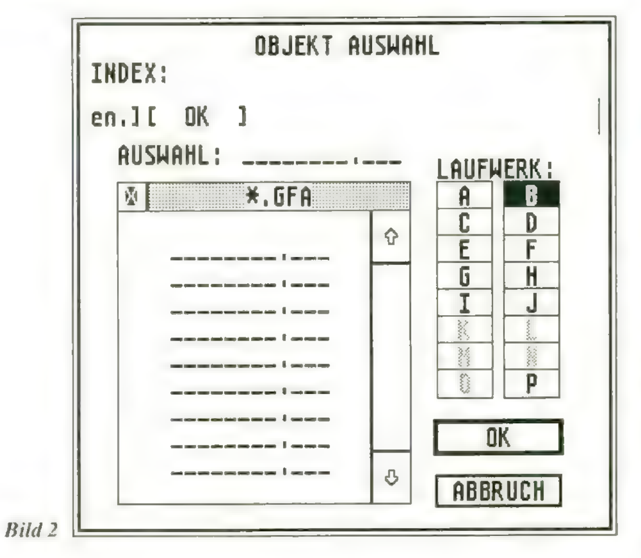


Bild 2

Andreas Precht, 5090 Leverkusen

Red.: Dieser Fall tritt immer dann auf, wenn die Optionen für die Box nicht korrekt vorbelegt sind. Da Sie "*.GFA" auswählen, gehe ich davon aus, daß Sie mit GFA-BASIC programmieren. In diesem Fall sieht der Aufruf folgendermaßen aus:

```
FILESELECT #"Erweiterungs-
text","\ *.GFA","", name$
```

"Erweiterungstext" wird erst ab TOS 1.4 nutzbar. Dieser Text wird statt "Objekt Auswahl" ausgegeben. In der nächsten Angabe liegt höchstwahrscheinlich Ihr Fehler, denn hier muß ein korrekter Pfad angegeben werden (beispielsweise "*.GFA"). Die nächste Option dient zur Vorbelegung eines bestimmten Dateinamens, in der letzten Variablen wird der Rückgabepfad abgelegt.

*

Jahresinhalts-verzeichnis

Bisher haben Sie jedes Jahr ein Jahresinhaltsverzeichnis mit einer kompletten Übersicht aller erschienenen Artikel und Themen veröffentlicht. Leider vermisse ich bis zu diesem Zeitpunkt die Liste von 1989. Ist sie einem Druckfehler zum Opfer geworden?

Burkhard Kugelmeier, 5870 Hemer

Red.: Viele Anfragen haben uns zu diesem Thema erreicht. Nicht die Liste, sondern den Eintrag in das Inhaltsverzeichnis der Januar-Ausgabe hat der Druckfehlerteufel erwischt. Das ausführliche Jahresinhaltsverzeichnis für 1989 ist in der Ausgabe 1/90 eingefügt und befindet sich dort zwischen den Seiten 18 und 19.



Dietmar Lorenz

MIDI-Software selber schreiben

Praktische Anleitung für Einsteiger und Profis
München 1990
GC Gunther Carstensen Verlag
234 Seiten inkl. Diskette
DM 46,-
ISBN 3-9802026-6-6

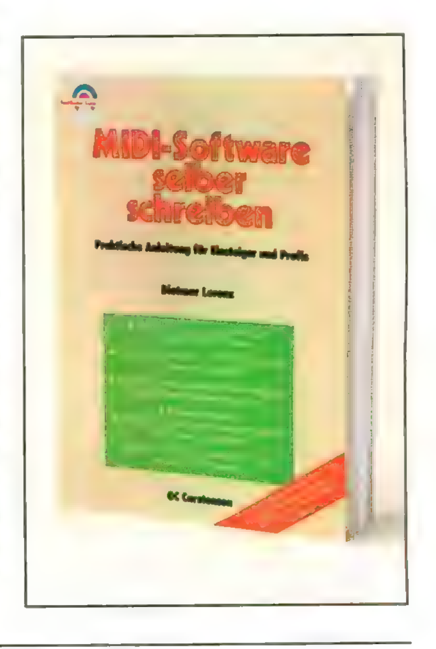
Unter den zahllosen Besitzern von MIDI-fähigen Keyboards (wenn Sie jetzt an das von Ihrem Computer denken, sind Sie hier falsch), Gitarren, Blasinstrumenten oder Kaffeemaschinen, gibt es doch einige, die den faustischen Wunsch verspüren, in die Tiefe der Materie sich zu begeben, um zu wissen, was die Töne wohl im Innersten zusammenhält. Es soll auch solche geben, denen das Angebot der MIDI-Software-Häuser immer noch nicht genügt und die tatsächlich glauben, sie könnten es besser...

Doch vor den Griff zur Tastatur (diesmal meine ich die des Computers) ist das Lernen gesetzt, das Lernen über MIDI und natürlich ein wenig Lernen über die Kunst des Programmierens. An willige, um genau zu sein, Lernwillige, wendet sich Dietmar Lorenz' Buch über Midi-Programmierung. Der Titel heißt zwar: 'Für Anfänger und Profis', aber das 'Profis' darf man getrost in die Schublade der Marketing-Gags stecken. Obwohl, es steht ja nicht dabei, was für Profis er meint, auch wenn Programmier- und Musikprofis wegfallen (die einen haben's nicht nötig, und die anderen lassen programmieren). Aber Spaß beiseite, für professionelle oder auch nur geübte Programmierer ist das Buch sicher nicht das Richtige. Es beschäftigt sich allein auf 70 von 234 Seiten mit einer Einführung in BASIC-Programmierung, sowohl in Omikron als auch in GFA (entsprechend 13,46 DM des Kaufpreises...). Das haben Profis doch nicht nötig, oder? Nein, Profis sind mit der Original-MIDI-Spezifikation sicher besser beraten.

Für Gelegenheits- und Hobby-Programmierer bietet 'Midi-Software selber schreiben' eine leicht lesbare und vor allem sehr genaue Einführung. Der Vorteil des Buches: Dieses eine ist genug, auch wenn Sie sehr wenige Vorkenntnisse haben. Sie können sich hinsetzen und experimentieren, nicht einmal tippen müssen Sie, denn alle Routinen und Beispiele werden auf Diskette mitgeliefert, jeweils in Omikron- und GFA-BASIC. Die Diskette ist lobenswerterweise auch noch PD und enthält zuguterletzt noch einen universellen Bankloader.

Eine Menge Projekte sind im Buch enthalten, vom MIDI-Monitor über einen einfachen Editor mit grafischer Bedienung (Hüllkurven-Editor!) bis hin zu einem einfachen Sequencer, und sogar rudimentäre Notengrafik gibt es. So richtig zum 'Reinriechen. Gefällt mir. Gut und verständlich erklärt und mit Beispielen in den verbreiteten BASIC-Dialekten. Das braucht zwar viel Platz, verhindert aber, daß der Anfänger über die Feinheiten der Sprachsyntax stolpert, die viel Zeit kosten und viel Frust bringen.

Kurzum: ein Buch, das für den interessierten Anfänger gut geeignet ist.



Immer up to date

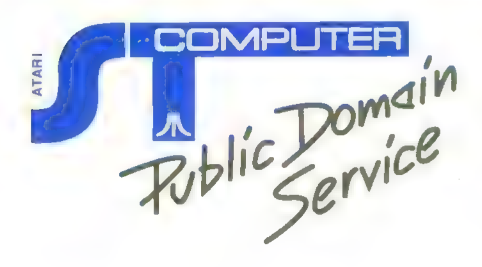
Mit dieser Sparte wollen wir allen unseren Lesern die Möglichkeit geben, sich über die neuesten Programm-Versionen zu informieren. Angegeben werden die aktuelle Versionsnummer, ein eventueller Kopierschutz, die Bildschirmauflösungen und der Speicherbedarf. Softwarefirmen ist es somit möglich, die ST-Computer-Leser über ihre Updates zu informieren.

Programmname	Version	Daten
Adimens ST	3.0	N HM
Adiprog SPC Modula	1.1	N HM
Aditalk ST	3.0	N HM
Adress ST / Check ST	1.0	N H
Afusoft Morse-Tutor	2.0	N HML
Afusoft Radio-Writer	1.0	N HML
Afusoft Radiofax plus	1.0	N HML 1M
AIDA	1.1	N HM
AnsiTerm	1.4	N
Assembler Tutorial	1.06	N HM
Banktransfer	1.0	N H
1st BASIC Tool	1.1	N HML
BTX/VTX-Manager	3.0	N H 1M
Calamus	1.09	N H 1M
Cashflow	1.0	N H 1M
Chips At Work	1.0	N HM
CIS-L&G	1.01	
Creator	1.1	N H
Diskus	1.07	N HM
dBMAN	5.10	N HML
Easytizer	1.0	N HM
Easy Rider Assembler	2.04	N HM
Easy Rider Reassembler	2.31	N HM
Edison	1.00	N HM
fibuMAN	3.0	N H
fibuSTAT	2.3	N H
Flexdisk	1.3	N HML
FM-Meßtechnik	1.0 b	N HM
FTL Modula-2	1.18	N HM
Gadget	1.2.5b	N H
GEMinterface ST	1.1	N HML
GFA Artist	1.0	N L
GFA Assembler	1.2	N HML
GFA BASIC 68881	1.3	N HML
GFA-BASIC Compiler	3.5	N HML
GFA-BASIC-Interpreter	3.5	N HML
GFA-Draft plus	3.0	N
GFA-Farb-Konverter	1.2	N H
GFA-Monochrom-Konverter	1.2	N ML
GFA Objekt	1.2	N HM
GFA-Starter	1.1	N HML
GFA Vektor	1.0	N
G+Plus	1.4	N HML
GrafStar	1.0	N H
Hänisch Modula-2	3.1	N HML
Hard Disk Accelerator	1.0	N HML
Hard Disk Sentry	1.10	
Hard Disk Toolkit	2.0	N HM
Harddisk Utility	2.2	N HM
Harlekin	1.0	N H 1M
Imagic	1.1	N HML
Intelligent Spooler	1.10	N HML
Interlink ST	1.69	N HM
Junior Prommer	2.33	N HM
K-Resource	2.0	N HM
Kleisterscheibe	2.2	N HM
Label ST	1.0	N HML
Laser C (Megamax)	2.1	N HML
1st Lektor	1.2	N HM
Lern ST	1.22	N HML
Link it GFA	1.1	N HML
Link it Omikron	2.0	N HML
MagicBOX ST	7.75a	N H 1M
Mathlib	3.0	N HM
Mega Paint II	2.30	N H 1M
Mega Paint II Professional	2.31	N H 1M
Megamax Modula 2	3.5	N HM
MGE Grafikkarte	2.0	N
MGP GAL-Prommer	1.03	N H
Micro C-Shell	2.70	N HM
Mr Print	3.0	N H
MT C-Shell	1.7	N HM 1M
Multidesk	1.82	N HML
Musix32	1.01	J H
NeoDesk	2.05	N HML
Omikron Assembler	1.86	N HML
Omikron BASIC-Compiler	3.06	N HML
Omikron BASIC 68881-Compiler	3.06	N HML
Omikron BASIC Interpreter	3.03	N HML
Omikron DRAW! 3.0	3.01	N HML
Omikron EasyGEM Lib	1.0	N HML
Omikron Maskeneditor	1.0	N HML
Omikron Midi-Lib	2.1	N HML
Omikron Numerik-Lib	1.2	N HML
Omikron Statistik-Lib	1.5	N HML
PAM's TERM 4014	3.012e	N H
PAM's TurboDisk	1.7	N HML
PAM's NET	1.1	N HML
PCB-layout	1.19	N H
PC ditto Euroversion	3.96	N HML
PegaFakt	1.3	N H
phs-BTX-Box	6.0	N HML 1M
phs-ST-Box	1.2	N HM
phs-Boxtalk	1.0	N HM 1M
phs-Boxedi	1.0	N HML 1M
phs-Cheapnet	1.2	N HM
Platon	1.45	N H
Prospero Pascal	2.151	N HML
Prospero Fortran	2.152	N HML
Prospero C-Compiler	1.142	N HML
Prospero Developers Toolkit	1.103	N HML
Protos	1.1	N H 1M
1st Proportional+	3.13	N HM
Quick_Dialog	1.0	N HM
ReProk	1.10	N H 1M
Revolver	1.1	N HML 1M
Search!	2.0	N HM
Signum! zwei	1.0	N H
Skylink	1.5	N H 1M
Skyplot+	4.1	J H 1M
Soundmachine II	1.0	N HM
SoundMerlin	1.01	N HM
SPC-Modula 2	2.0	N HML
Spectre 128	1.9	J HM
1st Speeder 2	1.0	N HML 1M
SPS ST	1.5	N H 1M
STAD	1.3+	N H
Steuer Tax 2.9	1.10	N HM
Steuer Tax 3.9	1.10	N HM
STop	1.1	N HM
ST Pascal plus	2.08	N HM
SuperScore	1.4	J H 1M
Tempus Editor	2.05	N HM
Theca Librarian	1.0	N HM
TIM	1.2	N H
TIM II	1.0	N H 1M
Transfile ST 1600	1.1	N HM
Transfile ST 850	1.1	N HM
Transfile ST plus	3.0	N HM
Turbo C	2.0	N HM
Turbo Debugger 1.0	1.0	N H
Turbo ST	1.8	N HML
UIS II + Hermes	2.5	
V Manager	3.02	N H
VSH Manager	1.0	N HML 1M
WERCS Resource-Editor	1.0	N HM

Irrtum vorbehalten!

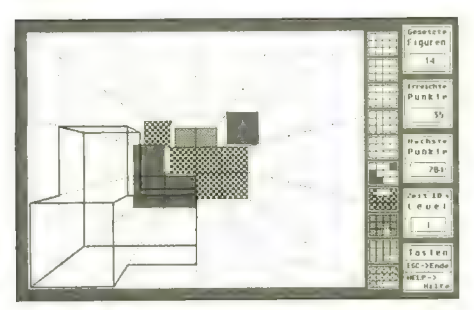
Daten-Legende : N = kein Kopierschutz, J = Kopierschutz, H = hohe Auflösung, M = mittlere Auflösung, L = niedrige Auflösung, 1M = mindestens 1 Megabyte

Inserentenverzeichnis



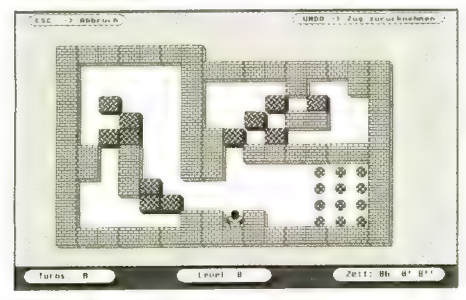
346 SPIELE

Endlich auch auf PD - eine dreidimensionale Tetris-Variante



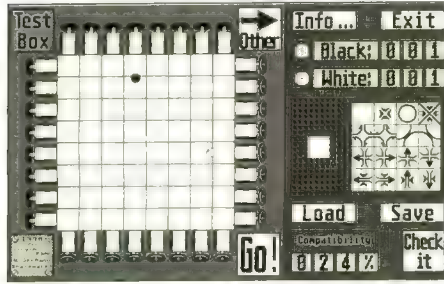
SETZ AB: hochinteressantes Tetris-Spiel in 3D. Es bietet verschiedene Schwierigkeitsgrade, ist leicht zu bedienen und wird Sie sicher stundenlang an den Bildschirm fesseln. Der Sourcecode liegt bei, hier kann man sich mal anschauen, wie solch ein Spiel programmiert wird. (s/w)

IRRGARTEN: sehr gutes, räumliches Irrgartenspiel. Die Größe und damit die Schwierigkeitsstufe läßt sich frei einstellen. Ziel des Spiel ist es, den Garten mit möglichst wenig Schritten zu verlassen, wobei man auf verschiedene Hilfen wie z.B. Kompaß, Zielpeiler, Positions- und Irrgartenanzeige zurückgreifen kann. Mit Sourcecode. (s/w)



PUSH BOX ist ein 'Kistenverschiebespiel', das mit seinen verschiedenen Levels doch einiges an Geistesarbeit fordert. Mit Sourcecode. (s/w)

TETRISS: eine weitere Tetris-Variante, diesmal in der legendären 2D-Version.



THE BOX ist ein nicht ganz einfaches Intelligenzspiel. Wer Spaß an Knobeleien hat, sollte diese Herausforderung annehmen.(s/w)

LASERSCHACH: Dies ist kein normales Schach-, sondern eher ein Mittelding zwischen Brett- und Ballerspiel. Es ist ein Brettspiel, bei dem man mit Hilfe von Spiegeln und Prismen versucht, den Gegner mattzusetzen. Hierbei steht einem noch eine Laserkanone zur Verfügung, mit deren Hilfe man die Figuren des Gegners abschießen kann. (s/w)

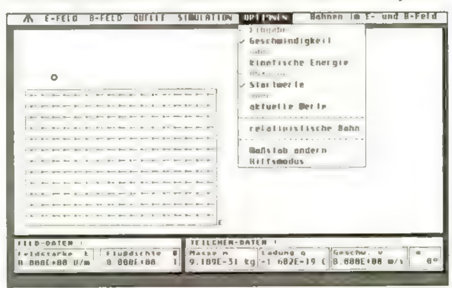
347

Nach Forth, Prolog, Lisp, Assembler, Modula 2 und C setzen wir die Serie der PD-Programmiersprachen fort.

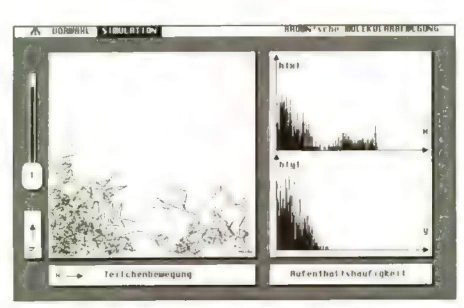
BC-FORTRAN77: ein vollständiges Fortran77-System. Es besteht aus Editor, Compiler, Linker und Bibilothek. Dieses System ist für nichtkommerzielle Anwendungen gedacht und eignet sich bestens zum Erlernen der Sprache bzw. für diejenigen, die ihre gerade erlernten Fähigkeiten nicht am überfüllten Uni-Rechner in die Praxis umsetzen wollen. Auf der Diskette ist ferner eine gute Anleitung zur Installation sowie zu Compiler- und Link-Optionen, Laufzeitsystem und Fehlermeldungen enthalten. (s/w)

348 PHYSIK

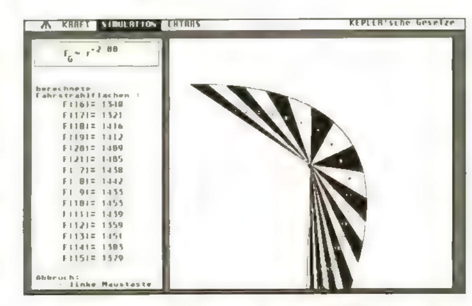
Auf dieser Diskette befinden sich Simulationsprogramme aus den verschiedensten Bereichen der Physik.



- Bewegungsbahnen geladener Teilchen in homogenen elektrischen und in Magnetfeldern

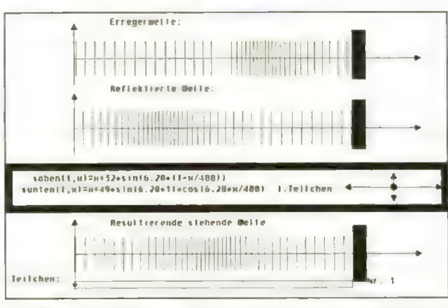


- Simulation verschiedener Grundeigenschaften idealer Gase



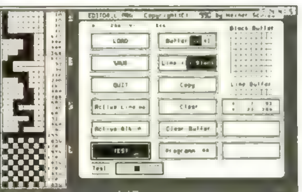
- Bewegungssimulation von bis zu 20 Körpern unter dem Einfluß ihrer sich überlagernden Gravitationsfelder

- Eigenschaften ferromagnetischer Stoffe
- Bewegung von Teilchen unter Einfluß beliebiger Kraftfelder
- Zerfall radioaktiver Präparate
- Brownsche Molekularbewegung



- Wellensimulation. Z.B Transversal- und Longitudinalwellen, Wasserwellen sowie die Beugung von Licht

349





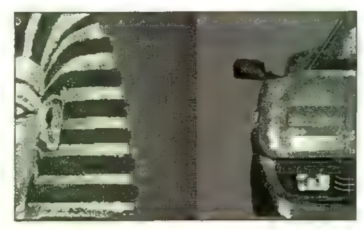
FLASH, RUNLIGHT: Programme zur Ansteuerung von Lampen und LEDs. Runlight ist ein Lauflichtprogramm, mit dem man über ein Interface die Verstärkerstufen ansteuern kann. Mit Flash kann man bis zu 8 verschiedene Lampen ansteuern. (s/w)

PORTFOLI: Datenübertragung vom ATARI Portfolio zum ATARI ST

WDR BILD: Bildkonverter vom WDR-Format ins ST-Screenformat

SPEEDWRITER: das etwas andere Textprogramm. Speedwriter zeichnet jeden Tastendruck auf (inkl. Korrekturen etc). Ihre Briefe versenden Sie also per Diskette, der Empfänger spielt ihn wieder genauso ab, wie Sie ihn geschrieben haben. Zusätzlich werden die Buchstaben mit Tönen unterlegt. (s/w)

350

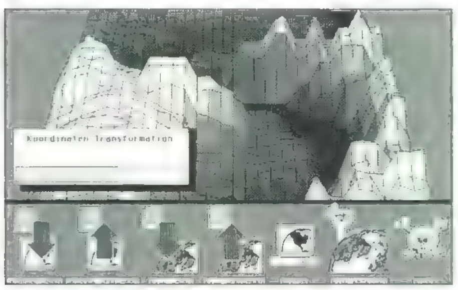


SHOWTIME: Haben Sie sich schon mal überlegt, wie es wäre, wenn man sich seine eigene "Diashow" erstellen könnte? Nun gibt es ein Programm, mit dem es

kinderleicht wird - SHOWTIME -. Mit diesem Programm ist es möglich, auf verschiedenste Weise Bilder ein- und auszublenden. Man kann auswählen, ob das Bild von oben, unten, rechts oder von links in den Monitor eingeblendet wird.



Ferner kann man Bilder überschneidend oder von der Mitte ausgehend auf den Monitor "projizieren". Komplexer Show-Generator samt Ablaufprogramm für eigenständige Demos. (s/w)

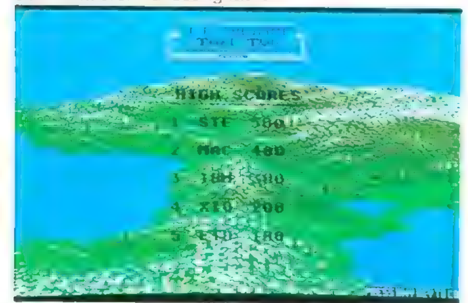


GEOSCAPE: dreidimensionale Darstellung geografischer Profile von Ländern oder geometrischen Körpern. Das Programm beinhaltet Algorithmen zur nachträglichen Beeinflussung (Zerklüften, Umstülpen) der Profile. (s/w)



351 FARBSPIELE

Nun kommt Farbe ins Spiel. Auf dieser PD-Diskette befinden sich ausschließlich Spiele, die nur in niedriger bzw. mittlerer Auflösung laufen.



TAKE TWO: Spiel ähnlich Shanghai. Möglichst schnell müssen zwei gleichpostierte Bausteine gefunden werden, um sie von der Spieloberfläche zu entfernen. Das Spiel geht über mehrere Levels und garantiert lange Spielfreude. (f)

PANIK: erweiterte Version von Panik (PD-315), die nur noch auf Farbe läuft. Ziel ist es, mit den verborgenen Schlüsseln den Weg aus engen Gängen zu finden, wobei man sich allerdings mit diversen Monstern herumschlagen muß. (f)

STATION: Adventurespiel, bei dem man mit seiner Mannschaft gegen feindliche Eindringlinge zu kämpfen hat. Ziel dieses Spiels ist es, in das feindliche Raumschiff einzudringen und es mit einem Team von Spezialisten zu zerstören. (f)

352 VAN GOGH



Das Farbmalprogramm VAN GOGH gab es bereits auf PD 68, aber zwischen diesen beiden Version gibt es riesige Unterschiede. VAN GOGH beherrscht neben unzähligen fantastischen Malfunktionen auch die Darstellung animierter Bewegungsabläufe. Somit läßt es sich auch für eigene Bildershows prima verwenden. VAN GOGH ist eines der besten Farbmalprogramme, die der ST bislang gesehen hat. Zu den überragenden Funktionen gehört u.a. eine Lupe mit absolut variablem Vergrößerungsfaktor. Jetzt neu in allen Auflösungen. (f)

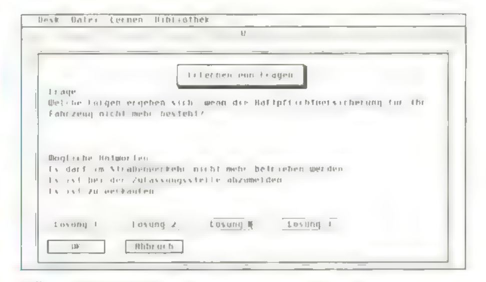
353 DATEN UND FINANZ

Auf dieser Diskette befinden sich die verschiedensten Programme, es fängt an mit dem Programm

PD GRADE: Unter diesem Namen verbergen sich mehrere Programme, mit denen man Berechnungen von Darlehen, Guthaben, Versicherungen und regelmäßigen Kosten anstellen kann. (s/w)

LCONTROL: zeigt, ob Sie sich in der für Sie günstigsten Steuerklasse befinden. Sie können somit überprüfen, ob es sich lohnt, die Steuerklasse zu wechseln.

AUSGABEN: Hilfsmittel zur Aufzeichnung der täglichen Ausgaben, damit man Ende des Monats weiß, wo das ganze liebe Geld hingekommen ist. (s/w)



FÜHRERSCHEIN: Lernprogramm zur Vorbereitung auf eine Führerscheinprüfung oder zur Kontrolle seines alten Theorie-Wissens.

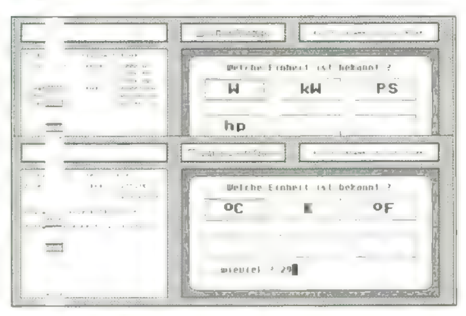
SCHACH DATO: Datenbank für Schachpartien. Verwaltung und Ausgabe von Schachpartien. Erfahrene Spieler werden das zu schätzen wissen. (s/w)

SBASE: flexible Datenbank für verschiedenste Anwendungen. SBASE ist frei konfigurierbar und sehr flexibel im Kontakt mit der Außenwelt.

354 MATHEMATIK

FUNKTION: berechnet Funktionsterme reeller Zahlen (s/w).

FORM: Programm zur Berechnung komplizierter mathematischer Ausdrücke, z.B. symbolische Berechnungen für Aufgaben der Mathematik, Physik oder den Maschinenbau. Wo andere Programme aufhören, beginnt FORM, doch Achtung: FORM ist sicherlich nichts für Anfänger. Samt 250seitiger englischer Anleitung auf Diskette. (s/w)



UMRECHEN: Berechnung div. Formeln mit den Einheiten Gewicht, Leistung, Länge, Fläche, Druck, Raum, Temperatur und Arbeit. (s/w)

UMWANDLER: Programm zur Umrechnung von verschiedenen Zahlensystemen (DEZ, HEX, DUAL)

355 PRINTING PRESS

Printing Press ist ein universelles Druckprogramm, in dem sehr viele Spezialprogramme zusammengefaßt sind. Mit diesem Programm können Sie Briefköpfe und Briefumschläge drucken. Sie können Bilder auf Poster oder als Diskettenetikett ausdrucken, und außerdem können Sie noch Banner und Grußkarten drucken. (s/w)

356 TOOLS SAMMLUNG

GIVE UP: Multi-Accessory, das Funktionen, wie zum Beispiel Kontrollfeld, Drucker, Uhr, Step-Rate, Bildschoner, 50Hz/60Hz-Umschaltung, Mousespeed, Hardcopy und vieles mehr enthält.

MEGAMATIC ist ebenfalls ein Multi-Accessory, das sich frei konfigurieren läßt, so daß nur die gewünschten Features gestartet werden.

JCLOCK: Uhrenprogramm

WERKZEUGKISTE: fantastisches Multi-Accessory mit 16 verschiedenen Funktionen (u.a. variable resetfeste RAM-Disk und Drucker-Spooler).
Die Bedienung von Werkzeugkiste ist sehr komfortabel, icongesteuert.

JAMES: ein Diener, der einige Aufgaben übernimmt.

X-UTILITY: Multitool, das die Unzulänglichkeiten des Betriebssystems ausgleicht, wie z.B. die Tastaturumbelegung.

DISKKATALOG: verwaltet Disketteninhalte in einer Datenbank

LOCK: Paßwortschutz für Rechner

CHOOSEBOOT: Boot-Wähler

DESKEDIT: Neue Icons für das Desktop (s/w)

GOODIES: Multitool mit vielen nützlichen Features, die einem hilfreich zur Seite stehen.

Edel, hilfreich und gut!

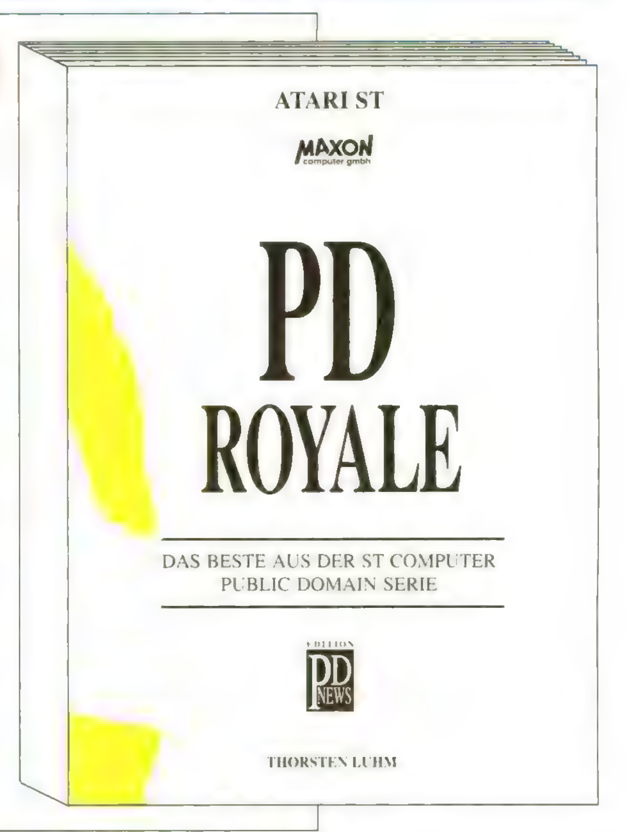
Public Domain - ein Begriff, der für viele hundert Programme und vor allem für eine Idee steht - eine Idee, die zum Ziel hat, selbstgeschriebene Programme der Allgemeinheit zur Verfügung zu stellen.

Die Redaktion der Zeitschrift ST-Computer begann Anfang 1986 damit, die so entstandenen Programme in ihrer Serie zu vereinen und erweckte damit das Kapitel Public Domain auf dem ATARI ST zum Leben.

Heute, 1990, enthält diese Sammlung über 350 Disketten mit ausgewählten PD- und Shareware-Programmen und ist damit ohne Zweifel die größte eigenständige PD-Sammlung. Sie enthält Programme aus allen nur erdenklichen Bereichen, sei es Wissenschaft, Spiele, Utilities, Grafik, Musik, Textverarbeitung oder gar Programmiersprachen und stellt daher eine wahre Fundgrube für jeden ST-Anwender dar. Daß es bei einer solchen Vielfalt nicht einfach ist, einen Überblick zu bekommen, dürfte niemanden wundern; gerade für Neuanwender des ST ist es fast unmöglich, aus dem reichhaltigen Angebot das Richtige herauszufinden. Daher gibt dieses Buch einen Überblick über die besten Programme und deren Einsatz in der Praxis.

Der Autor, Thorsten Luhm, beschäftigt sich seit langer Zeit mit PD-Programmen und kennt die ST Computer-Sammlung wie seine Westentasche. Dabei hat er viele Programme zu schätzen gelernt und an deren Entwicklung teilgenommen. Mit diesem Buch hat er sich die große Aufgabe vorgenommen, aus über 1000 Programmen die Leckerbissen herauszusuchen und dem Leser in leicht lesbarer und fachlich fundierter Weise zu präsentieren. Eine Aufgabe, die ihm, so glauben wir, rundum gelungen ist.

Bestellen Sie noch heute "PD ROYALE - Das Beste aus der ST Computer Public Domain Serie" entweder direkt bei MAXON Computer für DM 34.- inkl. Versandkosten oder über den Buchhandel (unverbindlicher Verkaufspreis DM 29.-, ISBN 3-927065-07-2).



UpDates

Wie auch im letzten Monat, erwähnen wir an dieser Stelle, welche Programme wir diesen Monat upgedatet haben.

PD 228 R_A_U_R 2.00 man kann nun auch das Bearbeiten und Zurückschreiben von RCS-Files im Intelformat durchführen.

PD 322 DIRECTORY-ANALYSER läuft jetzt auch im Farbmodus und statt der Pull-Down Menüs gibt es nun Dialogboxen.

PD 327 alle Programme wurden noch einmal überarbeitet und sind nun mit einem integrierten Virenschutzprogramm ausgestattet.

PD 341 HÜPFER bei diesem Programm kann man durch die Funktion "Ansehen - Plakat" die Einzelbilder verdeutlichen, da jetzt Bilderrahmen eingeblendet werden können.

PD 343 CASSETTE es wurden Fehler behoben und kleine optische Korrekturen durchgeführt, ferner kann jetzt jeder einzelne Song Steuerzeichen beinhalten.

PD 343 ZPRINT fehlerbereinigt

PD KOMPLETT

Die neue PD-News enthält eine Übersicht über die komplette PD-Serie der ST-Computer (PD 1-356).
Sie erhalten sie gegen Zusendung eines adressierten und mit DM 2.40 frankierten DIN A4 großen Rückumschlages.

MAXON-Computer • PD-NEWS
Industriestr. 26 • D-6236 Eschborn

Hier bitte Bild der neuen PD-NEWS!!

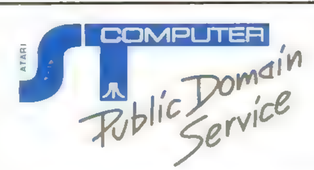
Machen Sie Mit!

Möchten Sie ein selbstgeschriebenes Programm in unsere PD-Sammlung geben, um es auch anderen Usern zugänglich zu machen? Kein Problem. Schicken Sie es uns auf einer Diskette zu, samt einer Bestätigung, daß es von Ihnen geschrieben wurde und frei von Rechten Dritter ist. Bei Fragen steht Ihnen die Redaktion gerne zur Verfügung.

MAXON Computer • ST-Computer PD
Industriestr. 26 • D-6236 Eschborn

ABKÜRZUNGEN

1MB = mind. 1MB Speicher notwendig
s/w = nur Monochrom; f = nur Farbe



DIREKT-VERSAND

Alle PD-Disketten unserer Sammlung gibt es nur direkt bei MAXON-Computer.

Um einen schnellen Versand zu ermöglichen, beachten Sie bitte folgende Punkte:

1. Schriftliche Bestellung

- Der Unkostenbeitrag für eine Diskette beträgt DM 10.-
- Hinzu kommen Versandkosten von DM 5.- (Ausland DM 10.-)
- Bezahlung per Scheck oder Nachnahme
- (Im Ausland nur Vorrauskasse möglich)
- Ab 5 Disketten entfallen die Versandkosten (DM 5.- bzw. DM 10.-)
- Bei Nachnahme zuzüglich DM 4,00 Nachnahmegebühr

2. Telefonische Bestellung

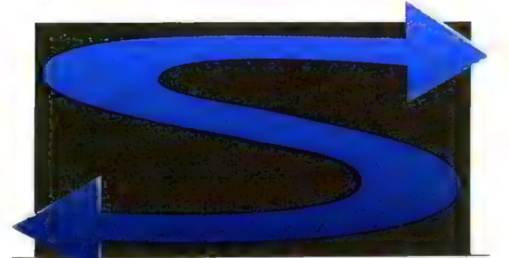
MAXON-Computer GmbH
"PD-Versand"
Tel.: 0 61 96 / 48 18 11
Fax: 0 61 96 / 4 18 85
Mo-Fr 9⁰⁰ - 13⁰⁰ und 14⁰⁰ -17⁰⁰ Uhr

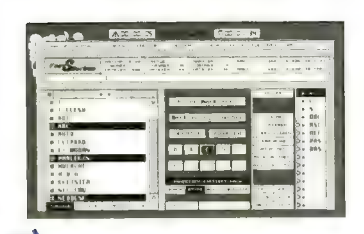
- Lieferung erfolgt per Nachnahme

Adresse:

MAXON-Computer GmbH
"PD ST-Computer"
Schwalbacher Strasse 52
D-6236 Eschborn

Nutzen Sie die PD-Karte in diesem Heft



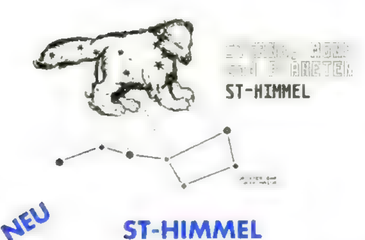


NEU

FastSectorBackup 4.0

Die Sicherheit Ihrer Daten sollte im Vordergrund stehen. FastSectorBackup ist das ideale Tool für diese Aufgabe. Zum einen bietet es ein Image-Backup, welches komplette Partitionen sichert, und zum anderen ein sehr flexibles FileBackup. Damit lassen sich einzelne Dateien, welche nach Wildcards, Datum, Archiv-Bit oder einfach per Mausklick markiert werden, sichern. Mittels FileBackup gesicherte Dateien können beliebig restored, also auch einzeln zurückkopiert werden.
Weiterhin bietet FastSectorBackup die Möglichkeit, mehrere backup-Vorgänge mit verschiedenen Markierungsarten (z.B. alle *.BAS-Dateien letzter Woche von C und alle nichtarchivierten *.DOC und *.TXT von E,...) in Batch-Dateien festzuhalten. Diese können dann automatisch ablaufen.
FastSectorBackup legt besonders Wert auf Datensicherheit, daher wird auf Diskettenfehler reagiert (nicht abgebrochen) und Prüfsummen gebildet. Weiterhin bietet das Programm die Möglichkeit, Dateien zu kopieren, umzubenennen, zu verschieben und zu komprimieren.

FastSectorBackup
SD 35 DM 25



NEU

ST-HIMMEL

Mit dem Programm kann der Anblick des Sternenhimmels für verschiedene Orte und Zeitpunkte berechnet werden. Ein ideales Programm für den Hobby-Astronomen.

Es zeigt:
- alle mit bloßem Auge (bei gutem Wetter) sichtbaren Sterne (~3000) mit Bezeichnungen, Helligkeiten und Entfernungen
- die mit bloßem Auge sichtb. Planeten
- den Mond mit seiner Phase
- die hellsten Sternhaufen und Nebel
- einen Kometen
- die Höhe der Sonne über oder unter dem Horizont
- die Namen der sichtbaren Planeten
- die verschiedenen Sternbilder
- den Tierkreis
- die Eigennamen von 190 Sternen (z.B. Großer Bär statt Ursa Maior)
- die Tag- und die Nachtseite der Erde auf einer Weltkarte

ST-Himmel ist besonders anwenderfreundlich, so kann beispielsweise der Standort auf einer zoombaren Welt- bzw. BRD/DDR-Karte angeklickt werden.

ST-HIMMEL
SD 38 DM 20.-

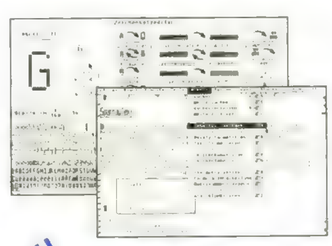


NEU

DATEI LOGIK

Datenbank, die einfache Handhabung und große Flexibilität miteinander vereint. So ist es für jedermann möglich, sich ohne große Anstrengung eine Datenbank nach seinen Vorstellungen aufzubauen, von A wie Adreßdatei über, wie Literaturverwaltung und P wie Plattenarchiv bis Z wie zoologische Sammlung. Mit Hilfe des integrierten Formular-Editors kann eine individuelle Abfragemaske erstellt, mit dem Etiketten-Editor das Layout von Aufklebern oder Karteikarten für jeden Aufgabenbereich festgelegt und mit der Mailmerge-Funktion mit den Daten auch Serienbriefe erstellt werden.

Datei Logik
SD 36 DM 20.-

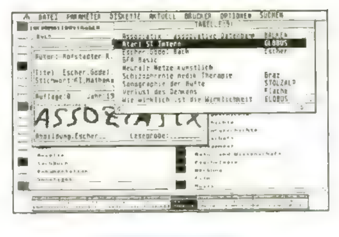


NEU

SparrowText

Exklusives Textverarbeitungssystem mit besonderen Leistungsmerkmalen. Neben der Darstellung aller Schriftarten auf dem Bildschirm beherrscht es verschiedene Zeilenabstände, Proportionalschrift im Blocksatz (variables Spacing), verschiedene Font-Größen und vor allem einen eigenen Bildschirmzeichensatz. Damit lassen sich Sonderzeichen entwerfen und auch an den Drucker schicken.
SparrowText unterstützt das Zeichnen von Linien und Rechtecken, Trennung, Textformatierung, automatische Erzeugung eines Inhaltsverzeichnisses und ist vor allem sehr schnell dabei.
Als besonderen Leckerbissen ermöglicht es Formularverarbeitung, die sich hervorragend zum Ausfüllen von Briefbögen, Adreßfeldern oder allgemeinen Formularen eignet. Die Eingabefelder lassen nach Wunsch auch Eingabebeschränkungen (z.B. nur Zahlen) zu und bieten daher die Möglichkeit, gewisse Felder miteinander aufzuaddieren. Weiterhin kann man diese Felder automatisch ausfüllen lassen, da SparrowText Daten von einer Datenbank importieren kann und diese in die Felder einträgt. Dadurch läßt sich das Programm für Serienbriefe, Zeugnisse oder gar Rechnungen, Mahnungen einsetzen.

SparrowText
SD 37 DM 25.-

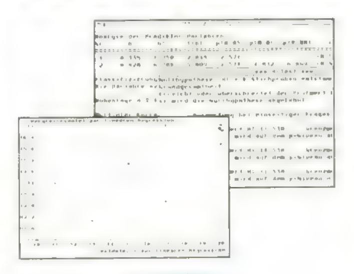


ASSOZIATIX

Assoziatix ist eine assoziativ-Muster orientierte Datenverwaltung, die es ermöglicht, aus einer großen Datenmenge bestimmte Gruppen auszufiltern und daraus dank schneller assoziativer Suche nach bestimmten Konstellationen Zusammenhänge zu finden (z.B. Rasterfandung).
Mit Hilfe des Formulareditors können die Eingabemasken leicht am Bildschirm gestaltet werden, sogar mit Grafikeinbindung.

Einige Besonderheiten:
- Passwortschutz, Export- und Importfunktion, Serienbriefe, Reportdokumentation, Statistische Berechnung numerischer Werte, Expertfunktion, Volltextsuche, Grafikeditor, Spiegeln, Drehen, Zoomen, Balken-, Linien und Kuchengrafik.

ASSOZIATIX (2 Disketten)
SD 27 a/b DM 30.-



STatiST

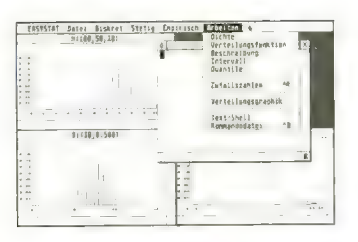
modulares Statistik-Programmpaket

STatiST ist ein umfangreiches Paket zur Auswertung statistischer Daten.
Es beinhaltet folgende 35 Testverfahren:

[illegible]

Zu jedem Prüfverfahren werden sämtliche Ergebnisse mit dem entsprechenden Wertungen und Kommentaren ausgegeben und, falls möglich, grafisch angezeigt.
STatiST eignet sich für sämtliche, z.B. im Studium erforderlichen statistischen Auswertungen und macht das zeitaufwendige Rechnen per Hand und das Arbeiten mit Tabellen überflüssig.

STATIST (2 Disketten)
SD 32a/b DM 30.-



EASYSTAT

Induktive Statistik

EASYSTAT dient der Errechnung und Veranschaulichung statistischer Verfahren. Dabei wird neben der beschreibenden Statistik vor allem die induktive Statistik berücksichtigt. Es eignet sich für alle Anwender der Statistik (Wirtschafts- und Sozialwissenschaftler, Techniker und Studenten). Mit EASYSTAT können Daten eingegeben, dargestellt, Kennzahlen berechnet, nach Zusammenhängen gesucht und Tests bzw. Intervallschätzungen durchgeführt werden. Von zentralen Verteilungen können Quantile (oder Pseudoquantile), Verteilungsfunktion und Wahrscheinlichkeitsfunktion (bzw. Dichte) berechnet werden. Z.B.: Geometrische, Hypergeometrische, (Negative) Binomial-, Normal- und Standardnormal-, Poisson-, t-, Chi-Quadrat-, F-, Exponential-, Erlang-n-, Weibull-, Beta-, Gamma-Verteilung.
EASYSTAT soll nicht zuletzt sehr abstrakte Dinge (z.B. statistische Tests) veranschaulichen helfen. Eine eingebaute einfache Kommandosprache ermöglicht es, Testprozeduren selbst zu schreiben. EASYSTAT beinhaltet einen speziell zugeschnittenen Editor und stellt ein On-Line-Hilfe-System zur Verfügung.

EASYSTAT
SD 31 DM 25.-

DAME

Computerumsetzung des alten Brettspiels, wobei der ST einen spielstarken Gegner darstellt. Die Figuren werden per Maus angewählt, die Züge protokolliert und analysiert. Verschiedene Spielstärken, Zugvorschläge, Laden und Speichern einer Partie, sowie verschiedene Spielvarianten dürfen nicht fehlen.

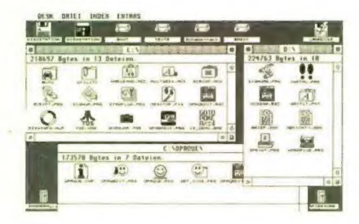
DAME
SD 29 DM 15.-

FUSSBALL

Tabellenverwaltung

FUSSBALL dient zur Verwaltung und Auswertung spezieller Informationen für Fußball-Fans. Das Programm erzeugt nach Eingabe der Ergebnisse die aktuelle Tabelle, und wertet die Daten gezielt statistisch aus u.a.: Besucherzahl und schnitt, Stadionauslastung, Rekordbesuch, bester Torschütze, Gesamtzahl gelbe/rote Karten..., Rundenstatistik, ... für gesamte Runde bzw. Saison, Team-bezogene Höchst- bzw. Tiefstwerte. FUSSBALL beinhaltet die kompletten Daten der Saison 89/90 (österreichische und Bundesliga), sowie die Paarungen der neuen Saison 90/91. Da sich das Punktesystem frei definieren läßt (z.B. auch 3 Punkte bei Sieg), lassen sich damit auch andere Sportarten verwalten.

FUSSBALL
SD 34 DM 15.-

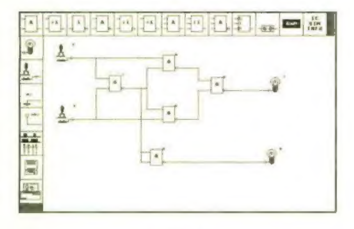


OPAQUE

Das Desktop mit neuem Gesicht

Wie wäre es mit einem zweckmäßigen und originellen Desktop? Opaque bietet die Möglichkeit, jedem Programm ein eigenes, sinnbezogenes Icon zuzuordnen. Auch die Laufwerke lassen sich ändern. Weiterhin kann man die Icons mit Wildcards definieren, z.B. *.BAS, *.TXT, *.ACC oder gar WORD????.PRG für verschiedene Programmversionen. Samt Icon-editor und über 100 Icons.

OPAQUE
SD 22 *DM 15.-*



ICSIM

Logik-Simulator

Das Programm simuliert das Verhalten von logischen Schaltungen. Bausteine und Verbindungen werden frei per Maus positioniert bzw. verbunden. Eine Schaltung läßt sich somit leicht austüfteln, testen und erst dann in die Praxis umsetzen.
Es sind die Logikbausteine nach DIN 40900 enthalten: AND, OR, NOT, NAND, NOR, XOR, RS-FF, KLEMME, LAMPE, SCHALTER, 0V und +5V.
Die Simulation wird als Impulsdiagramm oder Logiktabelle ausgegeben. Weiterhin liefert das Programm den Schaltplan und eine Liste der benötigten Bauteile.

ICSIM [1]
SD 25 *DM 20.-*

WÜRFELPOKER

Da sind die Würfelzocker gefragt. Würfelpoker kann man mit mehreren Personen (auch Computer) spielen. Sinn ist es, seine Würfe möglichst geschickt zu verteilen und gezielt nachzuwürfeln. Neben vielen Pärchen gilt es Straßen oder sonstige Kombinationen zu erzielen.

WÜRFELPOKER
SD 30 *DM 15.-*
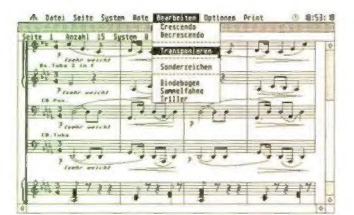
⇨ Sonderdisk-Bestellung

Sonderdisks können Sie telefonisch oder schriftlich bestellen, oder nutzen Sie einfach die Bestellkarte im Heft.

Bei Nachnahme zzgl. DM 4.- Gebühr, Versandkosten DM 5.- (Ausland DM 10.-)

MAXON Computer
Industriestr. 26
6236 Eschborn
Tel: 06196/481811

Prospekt gegen frankierten Rückumschlag



TRISTAN

Notensatzsystem

Für alle Musikfreunde, die nicht nur vom Blatt spielen, sondern auch aufs Blatt schreiben, bietet das Notensatzsystem TRISTAN die ideale Möglichkeit, ihre Noten professionell zu Papier zu bringen. Es lassen sich Partituren mit bis zu 100 Seiten mit max. 32 Notensystemen je Seite bearbeiten. Alle im klassischen Notensatz gebräuchlichen Zeichen, u.a. auch Schlagzeug- und Vorschlagnoten, lassen sich bequem mit der Maus edieren. Ebenfalls stehen mehrere Notenschlüssel, Sammelfahnen, Triller und Bindebögen zur Verfügung. Automatische Transponierfunktion. Ausdruck auf 9- und 24-Nadeldruckern in *maximaler* Druckerauflösung.

TRISTAN
SD 24 *DM 25.-*



Special Paint 2

Grafik de Luxe

Grafikprogramm der Extraklasse. Neben den vielen nützlichen Funktionen zeichnet sich Special Paint vor allem durch seine Geschwindigkeit, seine bequeme Bedienung und seine Kompatibilität zu bekannten Malprogrammen aus. Special Paint bietet umfangreiche Blockfunktionen, Lasso, superschnelle Lupe, Maskierungen, Clippen, schnelle Bieg-, Zerr- und Drehoptionen, Animation und vieles mehr. Clipboardunterstützung, umfangreiche Textfunktionen (ladbare Fonts, Blocksatz, Zeilenumbruch).

Special Paint [1]
SD 21 *DM 20.-*

DER MOTOR

Autotechnik

Der Motor erklärt mit zahlreichen Grafiken die Funktionsweise eines Verbrennungsmotors. Sehr anschaulich sind die beweglen Grafiken, die z.B. die Bewegung eines Kolbens und die Zündzeitpunkte deutlich machen. Das gezeigte Wissen wird zusätzlich in einem Quiz abgefragt. Jetzt mit **geregeltem Katalysator**!!

DER MOTOR [1]
SD 20 *DM 15.-*

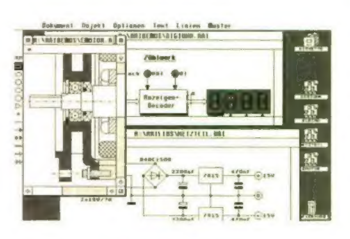
[1] nur für monochromen Monitor (SM 124)
[2] nur für Farbmonitor

HARDCOPY II

Die erste Farb-Hardcopy für den ST

Universelles Hardcopy-Tool. S/W- und Farb-Hardcopy auf *allen* Druckern in *allen* Größen, Screendump auf Disk, Formatkonvertierung, läuft als Accessory, einfachste Bedienung, optimale Druckqualität.

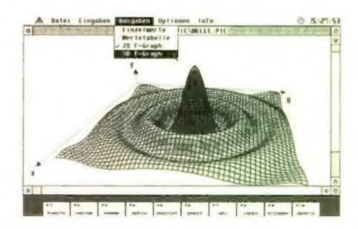
HARDCOPY II
SD 15 *DM 15.-*



ARIADNE

ARIADNE ist ein objektorientiertes Zeichenprogramm, d.h. Objekte können auch im Nachhinein ohne Auslösungsverlust verändert werden. Es bietet die Möglichkeit, jedes beliebige Grafikobjekt (mit Doppelklick) zu öffnen, worauf eine neue Zeichenebene bereitgestellt wird. Die Objekte auf dieser Ebene können dann wiederum geöffnet werden usw. Diese hierarchische Struktur eignet sich besonders zur Darstellung komplizierterer Dinge, z.B. Blockschaltbilder, Schaltungen etc.

ARIADNE [1]
SD 8 *DM 15,-*



FORMULA

2D-/ 3D-Plotter

Für mathematisch-wissenschaftliche Anwendung. Der eingebaute Formel-Interpreter beherrscht neben allen gängigen Operationen auch die Definition verschiedener Formeln in bestimmten Teilbereichen, logische Operationen und IF..THEN..ELSE. 3D-Grafiken lassen sich aus verschiedenen Blickrichtungen anzeigen und mit Schattierungen versehen.

FORMULA
SD 23 *DM 20.-*

PANDA

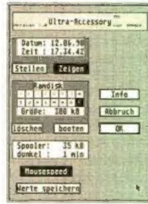
Der Farbemulator

Der Farbemulator simuliert die Farbauflösungen des ST auf einem monochromen Monitor (SM, 124,...). Dadurch kann man auch Farbspiele oder sonstige Farbprogramme laufen lassen, die sonst einen zweiten Monitor erfordern.

PANDA
SD 18 *DM 15,-*

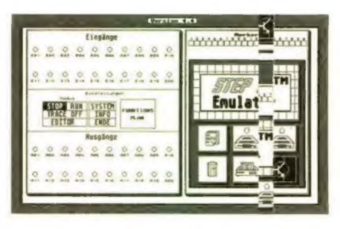
Ultra-Disk

RAM-Disk-Tool



Ultradisk ist eine ultraschnelle, größenveränderbare, resetfeste und reset-residente RAM-Disk. Die Größe und die Laufwerkskennung kann frei bestimmt werden, und das alles ohne Inhaltsverlust und ohne den Rechner neu zu booten. Weiterhin enthalten ist ein ultraschneller Drucker-Spooler, der dafür sorgt, daß Sie weiterarbeiten können, während der Rechner noch Daten an den Drucker schickt. Auch darf der Maus-Speeder, die Zeitanzeige und der Bildschirmschoner nicht fehlen.

ULTRA-DISK
SD 33 *DM 15.-*



SPS-Emulator

für programmierbare Steuerungen

Unser SPS-Emulator baut auf einem SIEMENS PG 605-Programmiergerät in STEP 5 auf. Mit ihm lassen sich SPS-Programme schreiben, auf Simulationsbasis austesten, laden, speichern, ändern, ausdrucken und als FUP (Funktionsplan mit logischen Gattern) ausgeben. Enthalten sind ein Editor, ein Interpreter und FUP-Generator.

SPS-Emulator [1]
SD 14 *DM 15,-*

Weitere Sonderdisks im Überblick

01	TOS 1.0	15,-
02	RCS 1.4	15,-
03	Extended VT52 [1]	15,-
04	Lovely Helper	15,-
05	Accessories	15,-
06	NIKI [1]	15,-
07	VirusEx	15,-
09	Legende [2]	15,-
10	Quinemac [1]	15,-
11	Patience [1]	15,-
12	MagicBox ST	15,-
13	Robotwar [1]	15,-
16	Easy Adress [1]	15,-
17	IconDesign	15,-
19	MAKI [1]	15,-
26	Hauskasse [1]	15,-
28	Master Etikett [1]	15,-

Diese Disketten können Sie auch weiterhin bestellen oder fordern Sie einfach unsere Info an.

- Sonderdisk, was ist das? -

Sonderdisks beinhalten Programme aus den verschiedensten Bereichen (z.B. Utilities, Grafik, Schulung, Spiele). Sonderdisks ermöglichen den Usern, qualitativ hochwertige Software zu einem kostengünstigen Preis zu erhalten. Im Preis ist eine Beteiligung der Autoren enthalten. Haben auch Sie ein Programm für diese Serie, so schreiben Sie uns.

MAXON Computer, Industriestr. 26
'Idee Sonderdisk', 6236 Eschborn

In der nächsten ST-Computer lesen Sie unter anderem

Quadratisch, praktisch, gut!

In der nächsten Ausgabe finden Sie natürlich auch einen allumfassenden Messebericht der diesjährigen ATARI-Messe. Natürlich vom TT mit neuem Desktop und 32 MHz, Script II, Calamus, Bildschirmtext und allem, was es neues gab und gibt. Ausgabe 10 verschweigt Ihnen nichts.

Auf die Plätze, fertig, FAX!

Endlich ist es soweit. Das erste FAX-Programm für den ST kommt auf den Markt. Welche Möglichkeiten Sie für unter 100,- DM haben, zeigt Ihnen unser Bericht. Wer hat nicht schon immer von einem eigenen FAX-Anschluß geträumt? Verwirklichen Sie ihn in der nächsten Ausgabe der ST Computer!

Schneller, höher, weiter!

Und wieder bieten wir Ihnen einen ROM-Patch an. Hardcopy mit dem NEC P6 ohne Druckertreiber, Bildschirmfarbe nach Wunsch, serielle Schnittstelle immer richtig gesetzt, Mausklick und Tastatur-Repeat nach Wunsch, jederzeit Uhranzeige auf dem Bildschirm, in hoher Auflösung sogar mit Wochentag und Datum. Sie glauben nicht, daß all diese Änderungen in ein geändertes TOS passen? Lassen Sie sich überzeugen von der Oktober-Ausgabe.

Die nächste ST Computer erscheint am Fr., den 28.9.90

Fragen an die Redaktion

Ein Magazin wie die ST-Computer zu erstellen, kostet sehr viel Zeit und Mühe. Da wir weiterhin vorhaben, die Qualität zu steigern, haben wir Redakteure ein großes Anliegen an Sie, liebe Leserinnen und Leser:

Bitte haben Sie Verständnis dafür, daß Fragen an die Redaktion nur **Donnerstags von 14^{00}-17^{00} Uhr** unter der Rufnummer 06196/481814 telefonisch beantwortet werden können.

Natürlich können wir Ihnen **keine** speziellen Einkauftips geben. Wenden Sie sich in diesem Fall bitte an einen Fachhändler. Wir können nur Fragen zur ST Computer beantworten.

Vielen Dank für Ihr Verständnis

Impressum ST Computer

Chefredakteur: Uwe Bärtels (UB)
Stellvertreter: Harald Egel (HE)

Redaktion:
Uwe Bärtels (UB)
Harald Egel (HE)
Joachim Merz (JM)
Martin Pittelkow (MP)

Redaktionelle Mitarbeiter:
C.Borgmeier (CBO)
Claus Brod (CB)
Ingo Brümmer (IB)
Derek dela Fuente (ddF)
Stefan Höhn (SH)
Raymund Hofmann (RH)
Dieter Kühner (DK)
Claus P. Lippert (CPL)
Chr. Schormann (CS)
R.Tolksdorf (RT)
Thomas Werner (TW)

Autoren dieser Ausgabe:
J.Bolt
D.Brockhaus
G. Ekart
A.Grunwald
U.Hax
U.Hervol
M.Kraft
A.Krieglmeier
R.Plamitzer
L. Reckmann (LR)
B.Rosenlecher
W.Rupflin
G.Schedel
C. Schmitz-Moormann
U.Seimet
P.Ubachs
R.Wiechert

Auslandskorrespondenz:
C.P.Lippert (Leitung), D.Dela Fuente (UK)

Redaktion: MAXON Computer GmbH
Postfach 59 69
Industriestr. 26
6236 Eschborn
Tel.: 0 61 96/48 18 14, FAX : 0 61 96/4 11 37

Verlag: Heim Fachverlag
Heidelberger Landstr. 194
6100 Darmstadt 13
Tel.: 0 61 51/5 60 57, FAX : 0 61 51/5 56 89 + 5 60 59

Verlagsleitung:
H.J.Heim

Anzeigenverkaufsleitung:
U.Heim

Anzeigenverkauf:
K.Margaritis

Anzeigenpreise:
nach Preisliste Nr.5, gültig ab 1.3.90
ISSN 0932-0385

Layout:
Martin Lowack, Manfred Zimmermann

Titelgestaltung:
Gunter Wenzel

Fotografie:
Martin Lowack

Illustration:
Manfred Zimmermann, Martin Lowack

Produktion:
K.H.Hoffmann, B.Kissner

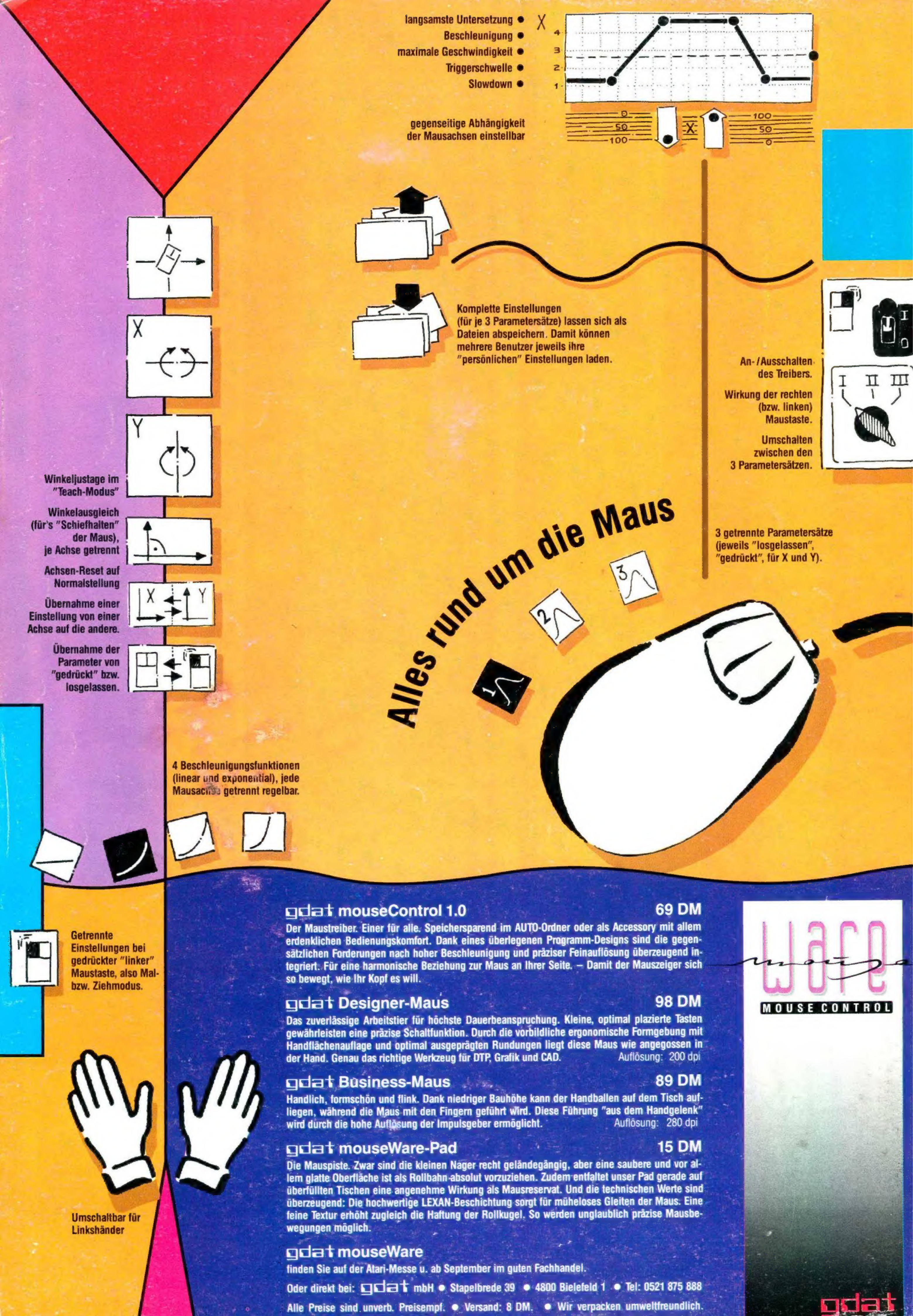
Druck:
Frotscher Druck GmbH

Lektorat:
V.Pfeiffer

Bezugsmöglichkeiten:
ATARI-Fachhandel, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag

ST Computer erscheint 11 x im Jahr
Einzelpreis: DM 8,-, ÖS 64,-, SFr 8,-
Jahresabonnement: DM 80,-
Europ. Ausland : DM 100,- Luftpost : DM 130,-
In den Preisen sind die gesetzliche MWSt. und die Zustellgebühren enthalten.

Manuskripteinsendungen:
Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne angenommen. Sie müssen frei von Rechten Dritter sein. Mit seiner Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung auf Datenträgern der MAXON Computer GmbH. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.



langsamste Untersetzung
Beschleunigung
maximale Geschwindigkeit
Triggerschwelle
Slowdown
gegenseitige Abhängigkeit der Mausachsen einstellbar
Komplette Einstellungen (für je 3 Parametersätze) lassen sich als Dateien abspeichern. Damit können mehrere Benutzer jeweils ihre "persönlichen" Einstellungen laden.
An- / Ausschalten des Treibers.
Wirkung der rechten (bzw. linken) Maustaste.
Umschalten zwischen den 3 Parametersätzen.
3 getrennte Parametersätze (jeweils "losgelassen", "gedrückt", für X und Y).
Alles rund um die Maus
Winkeljustage im "Teach-Modus"
Winkelausgleich (für's "Schiefhalten" der Maus), je Achse getrennt
Achsen-Reset auf Normalstellung
Übernahme einer Einstellung von einer Achse auf die andere.
Übernahme der Parameter von "gedrückt" bzw. losgelassen.
4 Beschleunigungsfunktionen (linear und exponentiell), jede Mausachse getrennt regelbar.
Getrennte Einstellungen bei gedrückter "linker" Maustaste, also Mal- bzw. Ziehmodus.
Umschaltbar für Linkshänder
gdat mouseControl 1.0 69 DM
Der Maustreiber. Einer für alle. Speichersparend im AUTO-Ordner oder als Accessory mit allem erdenklichen Bedienungskomfort. Dank eines überlegenen Programm-Designs sind die gegensätzlichen Forderungen nach hoher Beschleunigung und präziser Feinauflösung überzeugend integriert. Für eine harmonische Beziehung zur Maus an Ihrer Seite. – Damit der Mauszeiger sich so bewegt, wie Ihr Kopf es will.
gdat Designer-Maus 98 DM
Das zuverlässige Arbeitstier für höchste Dauerbeanspruchung. Kleine, optimal plazierte Tasten gewährleisten eine präzise Schaltfunktion. Durch die vorbildliche ergonomische Formgebung mit Handflächenauflage und optimal ausgeprägten Rundungen liegt diese Maus wie angegossen in der Hand. Genau das richtige Werkzeug für DTP, Grafik und CAD. Auflösung: 200 dpi
gdat Business-Maus 89 DM
Handlich, formschön und flink. Dank niedriger Bauhöhe kann der Handballen auf dem Tisch aufliegen, während die Maus mit den Fingern geführt wird. Diese Führung "aus dem Handgelenk" wird durch die hohe Auflösung der Impulsgeber ermöglicht. Auflösung: 280 dpi
gdat mouseWare-Pad 15 DM
Die Mauspiste. Zwar sind die kleinen Nager recht geländegängig, aber eine saubere und vor allem glatte Oberfläche ist als Rollbahn absolut vorzuziehen. Zudem entfaltet unser Pad gerade auf überfüllten Tischen eine angenehme Wirkung als Mausreservat. Und die technischen Werte sind überzeugend: Die hochwertige LEXAN-Beschichtung sorgt für müheloses Gleiten der Maus. Eine feine Textur erhöht zugleich die Haftung der Rollkugel. So werden unglaublich präzise Mausbewegungen möglich.
gdat mouseWare
finden Sie auf der Atari-Messe u. ab September im guten Fachhandel.
Oder direkt bei: gdat mbH • Stapelbrede 39 • 4800 Bielefeld 1 • Tel: 0521 875 888
Alle Preise sind unverb. Preisempf. • Versand: 8 DM. • Wir verpacken umweltfreundlich.
Ware
MOUSE CONTROL
gdat